Course: Deep Learning

Name: SAYEDPEDRAM HAERI BOROUJENI

Instructor: Dr. Feng Luo

Home Work: Number One

Part 3: Generalization

1. Import My Packages

In [1]:
```
import os
import torch
import torchvision
import torch.nn as nn
import torch.nn.functional as F
import torch.optim as optim
import torch.backends.cudnn as cudnn
import torchvision.transforms as transformtransforms
import torchvision.transforms.functional as TF
from torchvision import models
from torchsummary import summary
from torchvision import transforms
from torch.utils.data import Dataset, DataLoader
from torchvision.transforms import ToPILImage
from sklearn.decomposition import PCA
from mpl_toolkits.mplot3d import Axes3D
from tqdm import tqdm
import cv2
import copy
import math
import random
import numpy as np
import matplotlib.pyplot as plt
from PIL import Image
from glob import glob

os.environ["KMP_DUPLICATE_LIB_OK"]="TRUE"
TORCH_CUDA_ARCH_LIST="8.6"

Project_PATH = os.path.dirname(os.path.abspath('__file__'))
outputs_dir = Project_PATH + '/Desktop/Deep Learning HW1/'
model_path = Project_PATH + '/save_models/'
```

2. My Device

In [2]:
```
device_default = torch.cuda.current_device()
torch.cuda.device(device_default)
device = torch.device("cuda")
```

```
print("torch.cuda.is_available:", torch.cuda.is_available())
print("torch.cuda.device_count:", torch.cuda.device_count())
print("torch.cuda.current_device:", torch.cuda.current_device())
print("torch.cuda.get_device_name:", torch.cuda.get_device_name(device_default))
print("torch.version.cuda:", torch.version.cuda)
print("torch.version:", torch.__version__)
print("torch.cuda.arch_list:", torch.cuda.get_arch_list())
```

```
torch.cuda.is_available: True
torch.cuda.device_count: 1
torch.cuda.current_device: 0
torch.cuda.get_device_name: NVIDIA RTX A5000
torch.version.cuda: 11.3
torch.version: 1.11.0
torch.cuda.arch_list: ['sm_37', 'sm_50', 'sm_60', 'sm_61', 'sm_70', 'sm_75', 'sm_80',
'sm_86', 'compute_37']
```

3. My Model

In [3]:

```
class DNN_MNIST_N(nn.Module):
    def __init__(self, in_dim, hidden_1, hidden_2, hidden_3, out_dim):
        super(DNN_MNIST_N, self).__init__()
        self.layer1 = nn.Sequential(nn.Linear(in_dim, hidden_1),nn.BatchNorm1d(hidden_
        self.layer2 = nn.Sequential(nn.Linear(hidden_1, hidden_2),nn.BatchNorm1d(hidde
        self.layer3 = nn.Sequential(nn.Linear(hidden_2, hidden_3),nn.BatchNorm1d(hidde
        self.layer4 = nn.Sequential(nn.Linear(hidden_3, out_dim))

    def forward(self, x):
        x = self.layer1(x)
        x = self.layer2(x)
        x = self.layer3(x)
        x = self.layer4(x)
        return x

class DNN_MNIST_3(nn.Module):
    def __init__(self):
        super(DNN_MNIST_3, self).__init__()
        self.layer1 = nn.Sequential(nn.Linear(28*28, 256),nn.BatchNorm1d(256),nn.ReLU(
        self.layer2 = nn.Sequential(nn.Linear(256, 128),nn.BatchNorm1d(128),nn.ReLU(Tr
        self.layer3 = nn.Sequential(nn.Linear(128, 64),nn.BatchNorm1d(64),nn.ReLU(True
        self.layer4 = nn.Sequential(nn.Linear(64, 32),nn.BatchNorm1d(32),nn.ReLU(True)
        self.layer5 = nn.Sequential(nn.Linear(32, 10))

    def forward(self, x):
        x = self.layer1(x)
        x = self.layer2(x)
        x = self.layer3(x)
        x = self.layer4(x)
        x = self.layer5(x)
        return x

def standardization(x):
    x = np.array(x)
    x[np.isnan(x)] = 0
    return (x-np.mean(x))/np.std(x)
```

4. Train My Model

In [4]:
```
# 1. Training Function
def train_shuffle(model_name,
                  Epochs = 100,
                  Batch  = 2000,
                  Data_workers = 0,
                  LR = 0.01):

# 2. Initialization
    trainset = torchvision.datasets.MNIST(root='./data/',train=True,download=True,trar
    testset = torchvision.datasets.MNIST(root='./data/',train=False,download=True,trar
    random.shuffle(trainset.train_labels)
    trainloader = DataLoader(trainset, batch_size=Batch, shuffle=True, num_workers=Dat
    testloader  = DataLoader(testset,  batch_size=Batch, shuffle=True, num_workers=Dat
    print(trainset.classes)
    print(trainset.data.shape)
    print(testset.data.shape)

# 3. Initialization model
    torch.cuda.is_available()
    device = torch.device('cuda:0' if torch.cuda.is_available() else 'cpu')
    Model = model_name.to(device)

# 4. loss & optimizer
    criterion = nn.CrossEntropyLoss()
    optimizer = optim.SGD(Model.parameters(), lr=LR, momentum=0.9)
    scheduler = optim.lr_scheduler.StepLR(optimizer, step_size = 5, gamma = 0.8)

# 5. Training
    trainloss_list = []
    testloss_list  = []
    accuracy_list  = []
    lr_list = []

    for epoch in range(Epochs):
        Model.train()
        train_loss = 0.0
        for i, data in enumerate(trainloader):
            images, labels = data
            images = (images.view(-1, 28*28)).to(device)
            labels = labels.to(device)
            outputs = Model(images)
            loss = criterion(outputs, labels)
            optimizer.zero_grad()
            loss.backward()
            optimizer.step()
            train_loss += loss.item()
            total = (i+1)*Batch

# 6. Evaluating
        Model.eval()
        with torch.no_grad():
            test_loss = 0
            correct = 0
            total = 0
            for data in testloader:
                images, labels = data
```

```
                images = (images.view(-1, 28*28)).to(device)
                labels = labels.to(device)
                outputs = Model(images)
                loss = criterion(outputs, labels)
                test_loss += loss.item()
                _, pred = torch.max(outputs.data, 1)
                correct += (pred == labels).cpu().sum()
                total += labels.size(0)
            total = len(testloader.dataset)
            accuracy = 100.0*correct/total

# 7. Save loss
        scheduler.step()
        lr_list.append(optimizer.state_dict()['param_groups'][0]['lr'])
        trainloss_list.append(train_loss)
        testloss_list.append(test_loss)
        accuracy_list.append(accuracy)
        print('{}/{} Test set: Average loss: {:.4f}/{:.4f}, Accuracy: {}/{} ({:.2f}%)
                epoch, Epochs, train_loss,test_loss, correct, total, accuracy, lr_list

    return [trainloss_list,
            testloss_list,
            accuracy_list,
            lr_list]

[trainloss_list,testloss_list,accuracy_list,lr_list] = train_shuffle(model_name=DNN_MN
```

```
C:\Users\shaerib\Anaconda3\lib\site-packages\torchvision\datasets\mnist.py:65: UserWa
rning: train_labels has been renamed targets
  warnings.warn("train_labels has been renamed targets")
```

```
['0 - zero', '1 - one', '2 - two', '3 - three', '4 - four', '5 - five', '6 - six', '7
- seven', '8 - eight', '9 - nine']
torch.Size([60000, 28, 28])
torch.Size([10000, 28, 28])
0/100 Test set: Average loss: 70.5141/11.5723, Accuracy: 1047/10000 (10.47%) lr=0.01
1/100 Test set: Average loss: 68.9443/11.5841, Accuracy: 1104/10000 (11.04%) lr=0.01
2/100 Test set: Average loss: 68.4230/11.5959, Accuracy: 1149/10000 (11.49%) lr=0.01
3/100 Test set: Average loss: 68.0437/11.5819, Accuracy: 1205/10000 (12.05%) lr=0.01
4/100 Test set: Average loss: 67.6747/11.6196, Accuracy: 1212/10000 (12.12%) lr=0.008
5/100 Test set: Average loss: 67.2716/11.5437, Accuracy: 1349/10000 (13.49%) lr=0.008
6/100 Test set: Average loss: 66.9153/11.5734, Accuracy: 1247/10000 (12.47%) lr=0.008
7/100 Test set: Average loss: 66.5287/11.7064, Accuracy: 1153/10000 (11.53%) lr=0.008
8/100 Test set: Average loss: 66.1263/11.6283, Accuracy: 1252/10000 (12.52%) lr=0.008
9/100 Test set: Average loss: 65.6984/11.7695, Accuracy: 1118/10000 (11.18%) lr=0.006
4
10/100 Test set: Average loss: 65.1561/11.7701, Accuracy: 1143/10000 (11.43%) lr=0.00
64
11/100 Test set: Average loss: 64.6989/11.9059, Accuracy: 1109/10000 (11.09%) lr=0.00
64
12/100 Test set: Average loss: 64.2509/11.8368, Accuracy: 1142/10000 (11.42%) lr=0.00
64
13/100 Test set: Average loss: 63.7791/11.7805, Accuracy: 1199/10000 (11.99%) lr=0.00
64
14/100 Test set: Average loss: 63.2361/11.9188, Accuracy: 1179/10000 (11.79%) lr=0.00
512
15/100 Test set: Average loss: 62.5443/12.0349, Accuracy: 1139/10000 (11.39%) lr=0.00
512
16/100 Test set: Average loss: 62.0297/11.9799, Accuracy: 1206/10000 (12.06%) lr=0.00
512
17/100 Test set: Average loss: 61.4997/12.0050, Accuracy: 1240/10000 (12.40%) lr=0.00
512
18/100 Test set: Average loss: 60.9589/12.2385, Accuracy: 1130/10000 (11.30%) lr=0.00
512
19/100 Test set: Average loss: 60.4097/12.3856, Accuracy: 1106/10000 (11.06%) lr=0.00
4096000000000001
20/100 Test set: Average loss: 59.6406/12.5600, Accuracy: 898/10000 (8.98%) lr=0.0040
96000000000001
21/100 Test set: Average loss: 59.0589/12.4192, Accuracy: 1058/10000 (10.58%) lr=0.00
4096000000000001
22/100 Test set: Average loss: 58.5451/12.5918, Accuracy: 1020/10000 (10.20%) lr=0.00
4096000000000001
23/100 Test set: Average loss: 58.0578/12.5065, Accuracy: 1095/10000 (10.95%) lr=0.00
4096000000000001
24/100 Test set: Average loss: 57.5063/12.7699, Accuracy: 1014/10000 (10.14%) lr=0.00
32768000000000007
25/100 Test set: Average loss: 56.6909/12.7688, Accuracy: 1106/10000 (11.06%) lr=0.00
32768000000000007
26/100 Test set: Average loss: 56.1071/12.7026, Accuracy: 1105/10000 (11.05%) lr=0.00
32768000000000007
27/100 Test set: Average loss: 55.5908/13.0309, Accuracy: 909/10000 (9.09%) lr=0.0032
768000000000007
28/100 Test set: Average loss: 55.1310/13.0528, Accuracy: 981/10000 (9.81%) lr=0.0032
768000000000007
29/100 Test set: Average loss: 54.6057/13.0541, Accuracy: 1021/10000 (10.21%) lr=0.00
2621440000000001
30/100 Test set: Average loss: 53.7630/12.9301, Accuracy: 1160/10000 (11.60%) lr=0.00
2621440000000001
31/100 Test set: Average loss: 53.2588/13.3878, Accuracy: 952/10000 (9.52%) lr=0.0026
21440000000001
32/100 Test set: Average loss: 52.8147/13.4588, Accuracy: 970/10000 (9.70%) lr=0.0026
```

```
21440000000001
33/100 Test set: Average loss: 52.3658/13.3379, Accuracy: 1009/10000 (10.09%) lr=0.00
2621440000000001
34/100 Test set: Average loss: 51.8585/13.6198, Accuracy: 1014/10000 (10.14%) lr=0.00
2097152000000001
35/100 Test set: Average loss: 51.0954/13.4510, Accuracy: 1130/10000 (11.30%) lr=0.00
2097152000000001
36/100 Test set: Average loss: 50.6111/13.7514, Accuracy: 1002/10000 (10.02%) lr=0.00
2097152000000001
37/100 Test set: Average loss: 50.1849/13.8955, Accuracy: 995/10000 (9.95%) lr=0.0020
97152000000001
38/100 Test set: Average loss: 49.8588/13.8004, Accuracy: 1023/10000 (10.23%) lr=0.00
2097152000000001
39/100 Test set: Average loss: 49.4813/13.7643, Accuracy: 1026/10000 (10.26%) lr=0.00
1677721600000001
40/100 Test set: Average loss: 48.6571/14.1062, Accuracy: 955/10000 (9.55%) lr=0.0016
77721600000001
41/100 Test set: Average loss: 48.2446/14.1281, Accuracy: 966/10000 (9.66%) lr=0.0016
77721600000001
42/100 Test set: Average loss: 47.9234/14.2926, Accuracy: 972/10000 (9.72%) lr=0.0016
77721600000001
43/100 Test set: Average loss: 47.5842/14.4302, Accuracy: 933/10000 (9.33%) lr=0.0016
77721600000001
44/100 Test set: Average loss: 47.2793/14.3251, Accuracy: 1009/10000 (10.09%) lr=0.00
13421772800000008
45/100 Test set: Average loss: 46.5502/14.4015, Accuracy: 1011/10000 (10.11%) lr=0.00
13421772800000008
46/100 Test set: Average loss: 46.1752/14.5488, Accuracy: 950/10000 (9.50%) lr=0.0013
421772800000008
47/100 Test set: Average loss: 45.9139/14.5355, Accuracy: 1022/10000 (10.22%) lr=0.00
13421772800000008
48/100 Test set: Average loss: 45.6231/14.6176, Accuracy: 1027/10000 (10.27%) lr=0.00
13421772800000008
49/100 Test set: Average loss: 45.3409/14.8460, Accuracy: 973/10000 (9.73%) lr=0.0010
737418240000006
50/100 Test set: Average loss: 44.7843/14.7592, Accuracy: 998/10000 (9.98%) lr=0.0010
737418240000006
51/100 Test set: Average loss: 44.4624/14.8650, Accuracy: 945/10000 (9.45%) lr=0.0010
737418240000006
52/100 Test set: Average loss: 44.1957/14.7874, Accuracy: 1038/10000 (10.38%) lr=0.00
10737418240000006
53/100 Test set: Average loss: 43.9767/14.9945, Accuracy: 1018/10000 (10.18%) lr=0.00
10737418240000006
54/100 Test set: Average loss: 43.6847/14.9570, Accuracy: 1056/10000 (10.56%) lr=0.00
08589934592000006
55/100 Test set: Average loss: 43.1857/15.0585, Accuracy: 983/10000 (9.83%) lr=0.0008
589934592000006
56/100 Test set: Average loss: 42.9037/15.0963, Accuracy: 1027/10000 (10.27%) lr=0.00
08589934592000006
57/100 Test set: Average loss: 42.7084/15.1653, Accuracy: 991/10000 (9.91%) lr=0.0008
589934592000006
58/100 Test set: Average loss: 42.5032/15.2541, Accuracy: 1018/10000 (10.18%) lr=0.00
08589934592000006
59/100 Test set: Average loss: 42.2836/15.3456, Accuracy: 958/10000 (9.58%) lr=0.0006
871947673600005
60/100 Test set: Average loss: 41.8884/15.3931, Accuracy: 986/10000 (9.86%) lr=0.0006
871947673600005
61/100 Test set: Average loss: 41.6943/15.3757, Accuracy: 979/10000 (9.79%) lr=0.0006
871947673600005
62/100 Test set: Average loss: 41.5182/15.4335, Accuracy: 1003/10000 (10.03%) lr=0.00
```

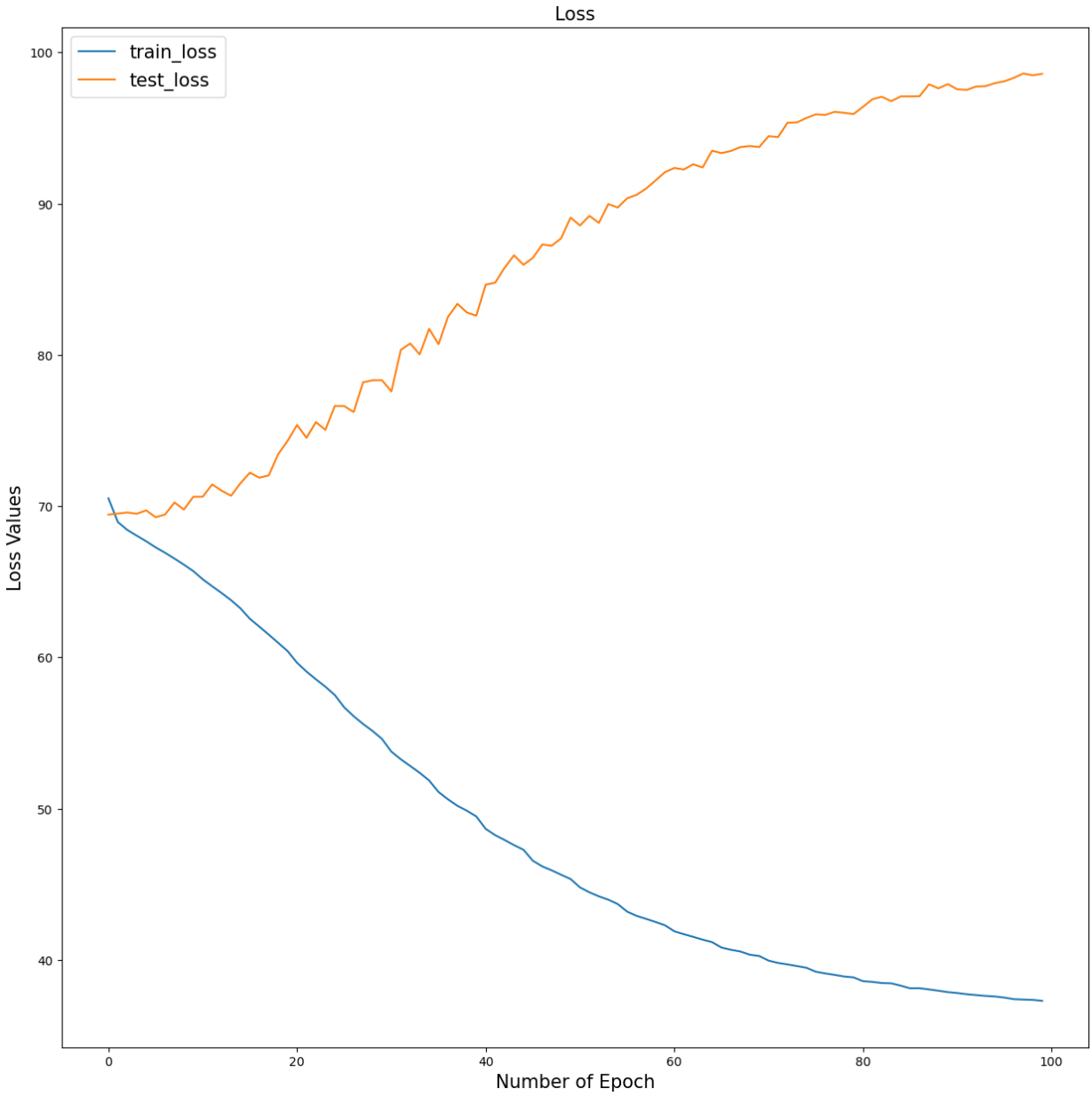
```
06871947673600005
63/100 Test set: Average loss: 41.3340/15.3979, Accuracy: 1002/10000 (10.02%) lr=0.00
06871947673600005
64/100 Test set: Average loss: 41.1678/15.5832, Accuracy: 963/10000 (9.63%) lr=0.0005
497558138880005
65/100 Test set: Average loss: 40.8117/15.5558, Accuracy: 997/10000 (9.97%) lr=0.0005
497558138880005
66/100 Test set: Average loss: 40.6627/15.5797, Accuracy: 985/10000 (9.85%) lr=0.0005
497558138880005
67/100 Test set: Average loss: 40.5509/15.6222, Accuracy: 1003/10000 (10.03%) lr=0.00
05497558138880005
68/100 Test set: Average loss: 40.3385/15.6341, Accuracy: 1004/10000 (10.04%) lr=0.00
05497558138880005
69/100 Test set: Average loss: 40.2543/15.6230, Accuracy: 1027/10000 (10.27%) lr=0.00
04398046511104004
70/100 Test set: Average loss: 39.9449/15.7430, Accuracy: 1007/10000 (10.07%) lr=0.00
04398046511104004
71/100 Test set: Average loss: 39.7934/15.7330, Accuracy: 994/10000 (9.94%) lr=0.0004
398046511104004
72/100 Test set: Average loss: 39.6970/15.8908, Accuracy: 991/10000 (9.91%) lr=0.0004
398046511104004
73/100 Test set: Average loss: 39.5872/15.8943, Accuracy: 987/10000 (9.87%) lr=0.0004
398046511104004
74/100 Test set: Average loss: 39.4747/15.9432, Accuracy: 992/10000 (9.92%) lr=0.0003
5184372088832035
75/100 Test set: Average loss: 39.2146/15.9836, Accuracy: 979/10000 (9.79%) lr=0.0003
5184372088832035
76/100 Test set: Average loss: 39.0999/15.9767, Accuracy: 990/10000 (9.90%) lr=0.0003
5184372088832035
77/100 Test set: Average loss: 39.0041/16.0113, Accuracy: 999/10000 (9.99%) lr=0.0003
5184372088832035
78/100 Test set: Average loss: 38.8947/16.0003, Accuracy: 986/10000 (9.86%) lr=0.0003
5184372088832035
79/100 Test set: Average loss: 38.8399/15.9867, Accuracy: 1001/10000 (10.01%) lr=0.00
02814749767106563
80/100 Test set: Average loss: 38.5882/16.0659, Accuracy: 995/10000 (9.95%) lr=0.0002
814749767106563
81/100 Test set: Average loss: 38.5408/16.1498, Accuracy: 989/10000 (9.89%) lr=0.0002
814749767106563
82/100 Test set: Average loss: 38.4645/16.1773, Accuracy: 981/10000 (9.81%) lr=0.0002
814749767106563
83/100 Test set: Average loss: 38.4456/16.1288, Accuracy: 987/10000 (9.87%) lr=0.0002
814749767106563
84/100 Test set: Average loss: 38.2921/16.1814, Accuracy: 999/10000 (9.99%) lr=0.0002
2517998136852504
85/100 Test set: Average loss: 38.1184/16.1814, Accuracy: 995/10000 (9.95%) lr=0.0002
2517998136852504
86/100 Test set: Average loss: 38.1226/16.1830, Accuracy: 985/10000 (9.85%) lr=0.0002
2517998136852504
87/100 Test set: Average loss: 38.0422/16.3141, Accuracy: 964/10000 (9.64%) lr=0.0002
2517998136852504
88/100 Test set: Average loss: 37.9563/16.2701, Accuracy: 974/10000 (9.74%) lr=0.0002
2517998136852504
89/100 Test set: Average loss: 37.8629/16.3163, Accuracy: 985/10000 (9.85%) lr=0.0001
8014398509482005
90/100 Test set: Average loss: 37.8042/16.2593, Accuracy: 1001/10000 (10.01%) lr=0.00
018014398509482005
91/100 Test set: Average loss: 37.7260/16.2530, Accuracy: 983/10000 (9.83%) lr=0.0001
8014398509482005
92/100 Test set: Average loss: 37.6698/16.2897, Accuracy: 997/10000 (9.97%) lr=0.0001
```

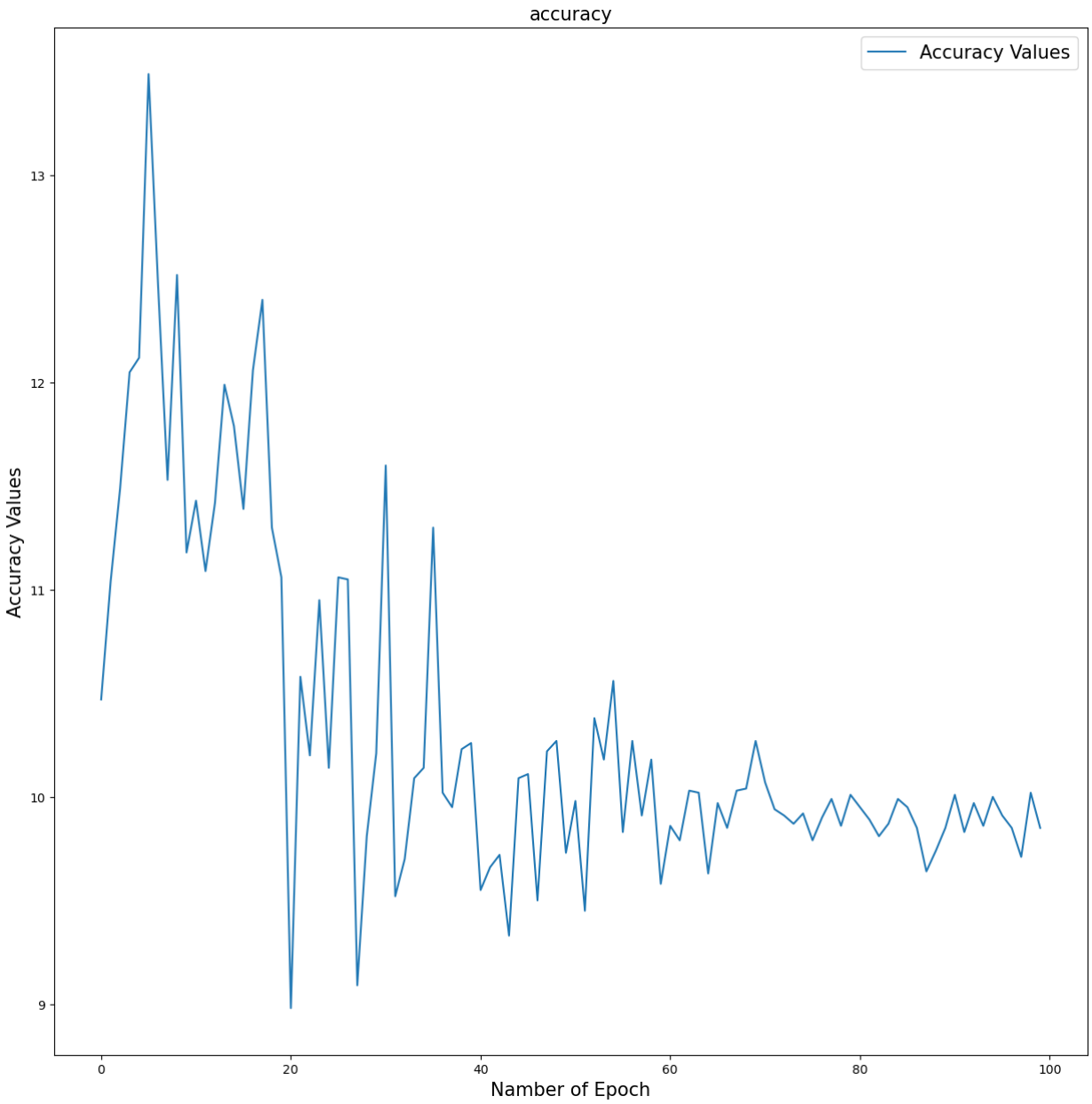
```
8014398509482005
93/100 Test set: Average loss: 37.6133/16.2941, Accuracy: 986/10000 (9.86%) lr=0.0001
8014398509482005
94/100 Test set: Average loss: 37.5766/16.3273, Accuracy: 1000/10000 (10.00%) lr=0.00
014411518807585605
95/100 Test set: Average loss: 37.5017/16.3481, Accuracy: 991/10000 (9.91%) lr=0.0001
4411518807585605
96/100 Test set: Average loss: 37.3982/16.3855, Accuracy: 985/10000 (9.85%) lr=0.0001
4411518807585605
97/100 Test set: Average loss: 37.3710/16.4337, Accuracy: 971/10000 (9.71%) lr=0.0001
4411518807585605
98/100 Test set: Average loss: 37.3505/16.4140, Accuracy: 1002/10000 (10.02%) lr=0.00
014411518807585605
99/100 Test set: Average loss: 37.2880/16.4302, Accuracy: 985/10000 (9.85%) lr=0.0001
1529215046068484
```

5. Ploting and Visualization

In [5]:

```
plt.figure(figsize=(15,15))
plt.plot(np.array(trainloss_list), label='train_loss')
plt.plot(np.array(testloss_list)*6, label='test_loss')
plt.xlabel('Number of Epoch',fontsize=15)
plt.ylabel('Loss Values',fontsize=15)
plt.title('Loss',fontsize=15)
plt.legend(fontsize=15)
plt.show()

plt.figure(figsize=(15,15))
plt.plot(accuracy_list, label='Accuracy Values')
plt.xlabel('Namber of Epoch',fontsize=15)
plt.ylabel('Accuracy Values',fontsize=15)
plt.title('accuracy',fontsize=15)
plt.legend(fontsize=15)
plt.show()
```


Loss
train_loss
test_loss
Loss Values
Number of Epoch
100
90
80
70
60
50
40
0
20
40
60
80
100



6. Generalization

In [6]:
```
def train_CIFAR10(model_name, Epochs=20, Batch=2000, Data_workers=0, LR=0.1):

# 1. Load Dataset
    transform = transforms.Compose([transforms.ToTensor(),transforms.Normalize((0.5,0.
    trainset = torchvision.datasets.CIFAR10(root='./data/',train=True,download=True,tr
    testset = torchvision.datasets.CIFAR10(root='./data/',train=False,download=True,tr
    trainloader = DataLoader(trainset, batch_size=Batch, shuffle=True, num_workers=Dat
    testloader  = DataLoader(testset,  batch_size=Batch, shuffle=True, num_workers=Dat
    print(trainset.classes)
    print(trainset.data.shape)
    print(testset.data.shape)
    torch.cuda.is_available()
    device = torch.device('cuda:0' if torch.cuda.is_available() else 'cpu')
    Model = model_name.to(device)
    num_param = sum(param.numel() for param in Model.parameters())
    print(model_name)
    print('Number of total parameters: ', num_param)
```

```
    criterion = nn.CrossEntropyLoss()
    optimizer = optim.SGD(Model.parameters(), lr=LR, momentum=0.9)
    scheduler = optim.lr_scheduler.StepLR(optimizer, step_size = 2, gamma = 0.8)

# 2. Training
    trainloss_list = []
    testloss_list  = []
    train_acc_list  = []
    test_acc_list = []
    lr_list = []
    for epoch in range(Epochs):
        Model.train()
        train_loss = 0.0
        train_correct = 0
        train_total = 0
        for i, data in enumerate(trainloader):
            images, labels = data
            images = (images.view(-1, 3*32*32)).to(device)
            labels = labels.to(device)
            outputs = Model(images)
            loss = criterion(outputs, labels)
            optimizer.zero_grad()
            loss.backward()
            optimizer.step()
            train_loss += loss.item()
            _, pred = torch.max(outputs.data, 1)
            train_correct += (pred == labels).cpu().sum()
        train_total = len(trainloader.dataset)
        train_acc = 100.0*train_correct/train_total

# 3. Evaluation
        Model.eval()
        with torch.no_grad():
            test_loss = 0
            test_correct = 0
            test_total = 0
            for data in testloader:
                images, labels = data
                images = (images.view(-1, 3*32*32)).to(device)
                labels = labels.to(device)
                outputs = Model(images)
                loss = criterion(outputs, labels)
                test_loss += loss.item()
                _, pred = torch.max(outputs.data, 1)
                test_correct += (pred == labels).cpu().sum()
            test_total = len(testloader.dataset)
            test_acc = 100.0*test_correct/test_total

# 4. Save Loss
        lr_list.append(optimizer.state_dict()['param_groups'][0]['lr'])
        trainloss_list.append(train_loss)
        testloss_list.append(test_loss)
        train_acc_list.append(train_acc)
        test_acc_list.append(test_acc)
        print('{}/{} Test set: Average loss: {:.4f}/{:.4f}, Accuracy: {}/{} ({:.2f}%)/
                epoch, Epochs, train_loss,test_loss, train_correct,test_correct, train

    return [trainloss_list, testloss_list,
            train_acc_list, test_acc_list,
            lr_list, num_param]
```

In [7]:
```
[trainloss_1,testloss_1,train_acc_1,test_acc_1,_,num_param_1] = train_CIFAR10(model_na
[trainloss_2,testloss_2,train_acc_2,test_acc_2,_,num_param_2] = train_CIFAR10(model_na
[trainloss_3,testloss_3,train_acc_3,test_acc_3,_,num_param_3] = train_CIFAR10(model_na
[trainloss_4,testloss_4,train_acc_4,test_acc_4,_,num_param_4] = train_CIFAR10(model_na
[trainloss_5,testloss_5,train_acc_5,test_acc_5,_,num_param_5] = train_CIFAR10(model_na
[trainloss_6,testloss_6,train_acc_6,test_acc_6,_,num_param_6] = train_CIFAR10(model_na
[trainloss_7,testloss_7,train_acc_7,test_acc_7,_,num_param_7] = train_CIFAR10(model_na
[trainloss_8,testloss_8,train_acc_8,test_acc_8,_,num_param_8] = train_CIFAR10(model_na
[trainloss_9,testloss_9,train_acc_9,test_acc_9,_,num_param_9] = train_CIFAR10(model_na
[trainloss_10,testloss_10,train_acc_10,test_acc_10,_,num_param_10] = train_CIFAR10(mod
```

```
Files already downloaded and verified
Files already downloaded and verified
['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'tr
uck']
(50000, 32, 32, 3)
(10000, 32, 32, 3)
DNN_MNIST_N(
  (layer1): Sequential(
    (0): Linear(in_features=3072, out_features=1, bias=True)
    (1): BatchNorm1d(1, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tru
e)
    (2): ReLU(inplace=True)
  )
  (layer2): Sequential(
    (0): Linear(in_features=1, out_features=2, bias=True)
    (1): BatchNorm1d(2, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tru
e)
    (2): ReLU(inplace=True)
  )
  (layer3): Sequential(
    (0): Linear(in_features=2, out_features=1, bias=True)
    (1): BatchNorm1d(1, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tru
e)
    (2): ReLU(inplace=True)
  )
  (layer4): Sequential(
    (0): Linear(in_features=1, out_features=10, bias=True)
  )
)
Number of total parameters:  3108
0/100 Test set: Average loss: 57.5763/11.2957, Accuracy: 6118/1693 (12.24%)/(16.93%)
lr=0.1
1/100 Test set: Average loss: 55.2135/10.8781, Accuracy: 8920/1848 (17.84%)/(18.48%)
lr=0.1
2/100 Test set: Average loss: 53.2609/10.5519, Accuracy: 9416/1844 (18.83%)/(18.44%)
lr=0.1
3/100 Test set: Average loss: 51.9132/10.3480, Accuracy: 9584/1938 (19.17%)/(19.38%)
lr=0.1
4/100 Test set: Average loss: 51.3125/10.2389, Accuracy: 9714/1951 (19.43%)/(19.51%)
lr=0.1
5/100 Test set: Average loss: 51.0788/10.1938, Accuracy: 9755/1965 (19.51%)/(19.65%)
lr=0.1
6/100 Test set: Average loss: 50.8922/10.1758, Accuracy: 9694/1986 (19.39%)/(19.86%)
lr=0.1
7/100 Test set: Average loss: 50.7954/10.1560, Accuracy: 9867/1969 (19.73%)/(19.69%)
lr=0.1
8/100 Test set: Average loss: 50.7148/10.1562, Accuracy: 9713/1994 (19.43%)/(19.94%)
lr=0.1
9/100 Test set: Average loss: 50.6898/10.1459, Accuracy: 9819/1973 (19.64%)/(19.73%)
lr=0.1
10/100 Test set: Average loss: 50.6862/10.1901, Accuracy: 9817/1952 (19.63%)/(19.52%)
lr=0.1
11/100 Test set: Average loss: 50.6619/10.1353, Accuracy: 9845/1893 (19.69%)/(18.93%)
lr=0.1
12/100 Test set: Average loss: 50.5694/10.1381, Accuracy: 9751/1867 (19.50%)/(18.67%)
lr=0.1
13/100 Test set: Average loss: 50.5524/10.1273, Accuracy: 9616/1902 (19.23%)/(19.02%)
lr=0.1
14/100 Test set: Average loss: 50.5333/10.1300, Accuracy: 9759/1965 (19.52%)/(19.65%)
lr=0.1
```

```
15/100 Test set: Average loss: 50.5308/10.1286, Accuracy: 9663/1886 (19.33%)/(18.86%)
lr=0.1
16/100 Test set: Average loss: 50.4985/10.1273, Accuracy: 9612/1974 (19.22%)/(19.74%)
lr=0.1
17/100 Test set: Average loss: 50.5056/10.1345, Accuracy: 9809/1918 (19.62%)/(19.18%)
lr=0.1
18/100 Test set: Average loss: 50.5286/10.1340, Accuracy: 9681/1933 (19.36%)/(19.33%)
lr=0.1
19/100 Test set: Average loss: 50.4715/10.1271, Accuracy: 9749/1964 (19.50%)/(19.64%)
lr=0.1
20/100 Test set: Average loss: 50.5274/10.1451, Accuracy: 9681/1947 (19.36%)/(19.47%)
lr=0.1
21/100 Test set: Average loss: 50.4820/10.1291, Accuracy: 9938/1853 (19.88%)/(18.53%)
lr=0.1
22/100 Test set: Average loss: 50.4998/10.1248, Accuracy: 9739/1973 (19.48%)/(19.73%)
lr=0.1
23/100 Test set: Average loss: 50.4490/10.1238, Accuracy: 9671/1937 (19.34%)/(19.37%)
lr=0.1
24/100 Test set: Average loss: 50.4637/10.1645, Accuracy: 9633/1908 (19.27%)/(19.08%)
lr=0.1
25/100 Test set: Average loss: 50.4726/10.1236, Accuracy: 9854/1932 (19.71%)/(19.32%)
lr=0.1
26/100 Test set: Average loss: 50.4188/10.1242, Accuracy: 9678/1954 (19.36%)/(19.54%)
lr=0.1
27/100 Test set: Average loss: 50.4425/10.1362, Accuracy: 9660/1897 (19.32%)/(18.97%)
lr=0.1
28/100 Test set: Average loss: 50.4491/10.1283, Accuracy: 9596/1972 (19.19%)/(19.72%)
lr=0.1
29/100 Test set: Average loss: 50.4387/10.1242, Accuracy: 9807/1969 (19.61%)/(19.69%)
lr=0.1
30/100 Test set: Average loss: 50.4293/10.2082, Accuracy: 9708/1944 (19.42%)/(19.44%)
lr=0.1
31/100 Test set: Average loss: 50.4825/10.1309, Accuracy: 9664/1928 (19.33%)/(19.28%)
lr=0.1
32/100 Test set: Average loss: 50.4273/10.1194, Accuracy: 9667/1990 (19.33%)/(19.90%)
lr=0.1
33/100 Test set: Average loss: 50.3904/10.1306, Accuracy: 9857/1941 (19.71%)/(19.41%)
lr=0.1
34/100 Test set: Average loss: 50.4163/10.1276, Accuracy: 9827/1952 (19.65%)/(19.52%)
lr=0.1
35/100 Test set: Average loss: 50.4177/10.1296, Accuracy: 9768/1974 (19.54%)/(19.74%)
lr=0.1
36/100 Test set: Average loss: 50.4172/10.1208, Accuracy: 9749/1982 (19.50%)/(19.82%)
lr=0.1
37/100 Test set: Average loss: 50.3682/10.1289, Accuracy: 9909/1996 (19.82%)/(19.96%)
lr=0.1
38/100 Test set: Average loss: 50.3672/10.1219, Accuracy: 9744/1961 (19.49%)/(19.61%)
lr=0.1
39/100 Test set: Average loss: 50.3706/10.1242, Accuracy: 9752/1873 (19.50%)/(18.73%)
lr=0.1
40/100 Test set: Average loss: 50.3990/10.1467, Accuracy: 9685/1892 (19.37%)/(18.92%)
lr=0.1
41/100 Test set: Average loss: 50.3768/10.1320, Accuracy: 9858/1870 (19.72%)/(18.70%)
lr=0.1
42/100 Test set: Average loss: 50.3767/10.1331, Accuracy: 9648/1882 (19.30%)/(18.82%)
lr=0.1
43/100 Test set: Average loss: 50.3615/10.1467, Accuracy: 9674/1972 (19.35%)/(19.72%)
lr=0.1
44/100 Test set: Average loss: 50.3515/10.1385, Accuracy: 9866/1939 (19.73%)/(19.39%)
lr=0.1
```

```
45/100 Test set: Average loss: 50.4170/10.1433, Accuracy: 9784/1970 (19.57%)/(19.70%)
lr=0.1
46/100 Test set: Average loss: 50.3626/10.1227, Accuracy: 9817/1987 (19.63%)/(19.87%)
lr=0.1
47/100 Test set: Average loss: 50.3563/10.1307, Accuracy: 9668/1995 (19.34%)/(19.95%)
lr=0.1
48/100 Test set: Average loss: 50.3307/10.1239, Accuracy: 9798/1913 (19.60%)/(19.13%)
lr=0.1
49/100 Test set: Average loss: 50.3628/10.1269, Accuracy: 9875/1908 (19.75%)/(19.08%)
lr=0.1
50/100 Test set: Average loss: 50.3565/10.1288, Accuracy: 9857/1880 (19.71%)/(18.80%)
lr=0.1
51/100 Test set: Average loss: 50.3294/10.1309, Accuracy: 9899/1908 (19.80%)/(19.08%)
lr=0.1
52/100 Test set: Average loss: 50.3514/10.1345, Accuracy: 9719/1944 (19.44%)/(19.44%)
lr=0.1
53/100 Test set: Average loss: 50.3477/10.1292, Accuracy: 9646/1938 (19.29%)/(19.38%)
lr=0.1
54/100 Test set: Average loss: 50.3384/10.1430, Accuracy: 9737/1900 (19.47%)/(19.00%)
lr=0.1
55/100 Test set: Average loss: 50.3441/10.1361, Accuracy: 9822/1943 (19.64%)/(19.43%)
lr=0.1
56/100 Test set: Average loss: 50.3371/10.1431, Accuracy: 9742/1879 (19.48%)/(18.79%)
lr=0.1
57/100 Test set: Average loss: 50.3269/10.1313, Accuracy: 9835/1917 (19.67%)/(19.17%)
lr=0.1
58/100 Test set: Average loss: 50.3388/10.1445, Accuracy: 9687/1991 (19.37%)/(19.91%)
lr=0.1
59/100 Test set: Average loss: 50.3615/10.1391, Accuracy: 9665/1947 (19.33%)/(19.47%)
lr=0.1
60/100 Test set: Average loss: 50.3245/10.1285, Accuracy: 9837/1975 (19.67%)/(19.75%)
lr=0.1
61/100 Test set: Average loss: 50.2925/10.1425, Accuracy: 9816/1891 (19.63%)/(18.91%)
lr=0.1
62/100 Test set: Average loss: 50.3336/10.1530, Accuracy: 9781/1917 (19.56%)/(19.17%)
lr=0.1
63/100 Test set: Average loss: 50.3710/10.1333, Accuracy: 9702/1974 (19.40%)/(19.74%)
lr=0.1
64/100 Test set: Average loss: 50.2910/10.1259, Accuracy: 9781/1952 (19.56%)/(19.52%)
lr=0.1
65/100 Test set: Average loss: 50.2964/10.1347, Accuracy: 9832/1923 (19.66%)/(19.23%)
lr=0.1
66/100 Test set: Average loss: 50.3133/10.1489, Accuracy: 9804/1923 (19.61%)/(19.23%)
lr=0.1
67/100 Test set: Average loss: 50.3167/10.1569, Accuracy: 9744/1878 (19.49%)/(18.78%)
lr=0.1
68/100 Test set: Average loss: 50.3359/10.1374, Accuracy: 9705/1938 (19.41%)/(19.38%)
lr=0.1
69/100 Test set: Average loss: 50.2908/10.1385, Accuracy: 9849/1943 (19.70%)/(19.43%)
lr=0.1
70/100 Test set: Average loss: 50.2756/10.1318, Accuracy: 9831/1935 (19.66%)/(19.35%)
lr=0.1
71/100 Test set: Average loss: 50.3014/10.1280, Accuracy: 9743/1999 (19.49%)/(19.99%)
lr=0.1
72/100 Test set: Average loss: 50.2775/10.1378, Accuracy: 9779/1972 (19.56%)/(19.72%)
lr=0.1
73/100 Test set: Average loss: 50.3197/10.1415, Accuracy: 9613/1937 (19.23%)/(19.37%)
lr=0.1
74/100 Test set: Average loss: 50.2968/10.1332, Accuracy: 9754/1950 (19.51%)/(19.50%)
lr=0.1
```

```
75/100 Test set: Average loss: 50.2870/10.1357, Accuracy: 9773/1991 (19.55%)/(19.91%)
lr=0.1
76/100 Test set: Average loss: 50.2798/10.1324, Accuracy: 9775/1918 (19.55%)/(19.18%)
lr=0.1
77/100 Test set: Average loss: 50.2665/10.1417, Accuracy: 9793/2001 (19.59%)/(20.01%)
lr=0.1
78/100 Test set: Average loss: 50.2936/10.1510, Accuracy: 9816/1894 (19.63%)/(18.94%)
lr=0.1
79/100 Test set: Average loss: 50.2906/10.1323, Accuracy: 9892/1860 (19.78%)/(18.60%)
lr=0.1
80/100 Test set: Average loss: 50.3328/10.1413, Accuracy: 9914/1949 (19.83%)/(19.49%)
lr=0.1
81/100 Test set: Average loss: 50.2959/10.1331, Accuracy: 9701/2005 (19.40%)/(20.05%)
lr=0.1
82/100 Test set: Average loss: 50.2608/10.1460, Accuracy: 9702/1997 (19.40%)/(19.97%)
lr=0.1
83/100 Test set: Average loss: 50.3078/10.1435, Accuracy: 9691/1878 (19.38%)/(18.78%)
lr=0.1
84/100 Test set: Average loss: 50.2758/10.1381, Accuracy: 9774/1939 (19.55%)/(19.39%)
lr=0.1
85/100 Test set: Average loss: 50.2742/10.1359, Accuracy: 9775/1877 (19.55%)/(18.77%)
lr=0.1
86/100 Test set: Average loss: 50.2729/10.1418, Accuracy: 9798/1978 (19.60%)/(19.78%)
lr=0.1
87/100 Test set: Average loss: 50.2689/10.1837, Accuracy: 9925/1925 (19.85%)/(19.25%)
lr=0.1
88/100 Test set: Average loss: 50.3073/10.1427, Accuracy: 9734/1935 (19.47%)/(19.35%)
lr=0.1
89/100 Test set: Average loss: 50.3055/10.1404, Accuracy: 9686/1945 (19.37%)/(19.45%)
lr=0.1
90/100 Test set: Average loss: 50.2894/10.1330, Accuracy: 9776/1954 (19.55%)/(19.54%)
lr=0.1
91/100 Test set: Average loss: 50.2947/10.1501, Accuracy: 9760/2003 (19.52%)/(20.03%)
lr=0.1
92/100 Test set: Average loss: 50.2723/10.1406, Accuracy: 9934/1884 (19.87%)/(18.84%)
lr=0.1
93/100 Test set: Average loss: 50.2851/10.1552, Accuracy: 9844/1943 (19.69%)/(19.43%)
lr=0.1
94/100 Test set: Average loss: 50.3094/10.1366, Accuracy: 9805/1926 (19.61%)/(19.26%)
lr=0.1
95/100 Test set: Average loss: 50.2527/10.1416, Accuracy: 9918/1997 (19.84%)/(19.97%)
lr=0.1
96/100 Test set: Average loss: 50.2488/10.1355, Accuracy: 9851/1976 (19.70%)/(19.76%)
lr=0.1
97/100 Test set: Average loss: 50.2438/10.1392, Accuracy: 9918/1938 (19.84%)/(19.38%)
lr=0.1
98/100 Test set: Average loss: 50.2522/10.1399, Accuracy: 9794/1956 (19.59%)/(19.56%)
lr=0.1
99/100 Test set: Average loss: 50.2811/10.1503, Accuracy: 9835/1964 (19.67%)/(19.64%)
lr=0.1
Files already downloaded and verified
Files already downloaded and verified
['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'tr
uck']
(50000, 32, 32, 3)
(10000, 32, 32, 3)
DNN_MNIST_N(
  (layer1): Sequential(
    (0): Linear(in_features=3072, out_features=2, bias=True)
    (1): BatchNorm1d(2, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tru
```

```
e)
    (2): ReLU(inplace=True)
  )
  (layer2): Sequential(
    (0): Linear(in_features=2, out_features=4, bias=True)
    (1): BatchNorm1d(4, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tru
e)
    (2): ReLU(inplace=True)
  )
  (layer3): Sequential(
    (0): Linear(in_features=4, out_features=2, bias=True)
    (1): BatchNorm1d(2, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tru
e)
    (2): ReLU(inplace=True)
  )
  (layer4): Sequential(
    (0): Linear(in_features=2, out_features=10, bias=True)
  )
)
Number of total parameters:  6214
0/100 Test set: Average loss: 57.8239/11.2148, Accuracy: 6475/1570 (12.95%)/(15.70%)
lr=0.1
1/100 Test set: Average loss: 55.4905/10.8812, Accuracy: 8392/1885 (16.78%)/(18.85%)
lr=0.1
2/100 Test set: Average loss: 53.7912/10.5469, Accuracy: 9501/2098 (19.00%)/(20.98%)
lr=0.1
3/100 Test set: Average loss: 52.0343/10.2407, Accuracy: 10782/2258 (21.56%)/(22.58%)
lr=0.1
4/100 Test set: Average loss: 50.5467/10.0419, Accuracy: 12196/2413 (24.39%)/(24.13%)
lr=0.1
5/100 Test set: Average loss: 49.5947/9.8814, Accuracy: 12699/2540 (25.40%)/(25.40%)
lr=0.1
6/100 Test set: Average loss: 48.9415/9.7698, Accuracy: 12896/2551 (25.79%)/(25.51%)
lr=0.1
7/100 Test set: Average loss: 48.4714/9.6772, Accuracy: 13089/2577 (26.18%)/(25.77%)
lr=0.1
8/100 Test set: Average loss: 48.2072/9.6381, Accuracy: 13190/2628 (26.38%)/(26.28%)
lr=0.1
9/100 Test set: Average loss: 48.0395/9.6194, Accuracy: 13272/2677 (26.54%)/(26.77%)
lr=0.1
10/100 Test set: Average loss: 47.9366/9.6008, Accuracy: 13325/2589 (26.65%)/(25.89%)
lr=0.1
11/100 Test set: Average loss: 47.8077/9.5621, Accuracy: 13341/2644 (26.68%)/(26.44%)
lr=0.1
12/100 Test set: Average loss: 47.7266/9.5342, Accuracy: 13360/2618 (26.72%)/(26.18%)
lr=0.1
13/100 Test set: Average loss: 47.6276/9.5488, Accuracy: 13516/2603 (27.03%)/(26.03%)
lr=0.1
14/100 Test set: Average loss: 47.6142/9.5810, Accuracy: 13465/2601 (26.93%)/(26.01%)
lr=0.1
15/100 Test set: Average loss: 47.6041/9.5717, Accuracy: 13496/2656 (26.99%)/(26.56%)
lr=0.1
16/100 Test set: Average loss: 47.5405/9.5154, Accuracy: 13677/2634 (27.35%)/(26.34%)
lr=0.1
17/100 Test set: Average loss: 47.5058/9.5411, Accuracy: 13634/2633 (27.27%)/(26.33%)
lr=0.1
18/100 Test set: Average loss: 47.4483/9.5563, Accuracy: 13799/2693 (27.60%)/(26.93%)
lr=0.1
19/100 Test set: Average loss: 47.4921/9.5946, Accuracy: 13894/2675 (27.79%)/(26.75%)
lr=0.1
```

```
20/100 Test set: Average loss: 47.4170/9.5218, Accuracy: 13743/2754 (27.49%)/(27.54%)
lr=0.1
21/100 Test set: Average loss: 47.3696/9.5257, Accuracy: 13752/2722 (27.50%)/(27.22%)
lr=0.1
22/100 Test set: Average loss: 47.3102/9.4781, Accuracy: 13977/2735 (27.95%)/(27.35%)
lr=0.1
23/100 Test set: Average loss: 47.3018/9.5120, Accuracy: 13877/2643 (27.75%)/(26.43%)
lr=0.1
24/100 Test set: Average loss: 47.2934/9.5079, Accuracy: 13996/2797 (27.99%)/(27.97%)
lr=0.1
25/100 Test set: Average loss: 47.2679/9.5163, Accuracy: 13986/2820 (27.97%)/(28.20%)
lr=0.1
26/100 Test set: Average loss: 47.2192/9.5179, Accuracy: 14127/2751 (28.25%)/(27.51%)
lr=0.1
27/100 Test set: Average loss: 47.2698/9.4837, Accuracy: 14035/2729 (28.07%)/(27.29%)
lr=0.1
28/100 Test set: Average loss: 47.2328/9.4867, Accuracy: 14112/2788 (28.22%)/(27.88%)
lr=0.1
29/100 Test set: Average loss: 47.1442/9.5012, Accuracy: 14299/2799 (28.60%)/(27.99%)
lr=0.1
30/100 Test set: Average loss: 47.1306/9.4814, Accuracy: 14293/2754 (28.59%)/(27.54%)
lr=0.1
31/100 Test set: Average loss: 47.1396/9.5139, Accuracy: 14304/2789 (28.61%)/(27.89%)
lr=0.1
32/100 Test set: Average loss: 47.1102/9.4966, Accuracy: 14146/2732 (28.29%)/(27.32%)
lr=0.1
33/100 Test set: Average loss: 47.0685/9.4728, Accuracy: 14283/2788 (28.57%)/(27.88%)
lr=0.1
34/100 Test set: Average loss: 47.0713/9.4822, Accuracy: 14359/2718 (28.72%)/(27.18%)
lr=0.1
35/100 Test set: Average loss: 47.0878/9.4989, Accuracy: 14335/2828 (28.67%)/(28.28%)
lr=0.1
36/100 Test set: Average loss: 47.0546/9.4900, Accuracy: 14355/2719 (28.71%)/(27.19%)
lr=0.1
37/100 Test set: Average loss: 47.0107/9.5552, Accuracy: 14390/2735 (28.78%)/(27.35%)
lr=0.1
38/100 Test set: Average loss: 47.0358/9.4892, Accuracy: 14465/2795 (28.93%)/(27.95%)
lr=0.1
39/100 Test set: Average loss: 47.0143/9.7213, Accuracy: 14404/2684 (28.81%)/(26.84%)
lr=0.1
40/100 Test set: Average loss: 47.0210/9.5277, Accuracy: 14486/2703 (28.97%)/(27.03%)
lr=0.1
41/100 Test set: Average loss: 46.9928/9.5561, Accuracy: 14446/2774 (28.89%)/(27.74%)
lr=0.1
42/100 Test set: Average loss: 46.9584/9.4916, Accuracy: 14361/2813 (28.72%)/(28.13%)
lr=0.1
43/100 Test set: Average loss: 46.9463/9.4806, Accuracy: 14556/2830 (29.11%)/(28.30%)
lr=0.1
44/100 Test set: Average loss: 46.9086/9.4808, Accuracy: 14550/2750 (29.10%)/(27.50%)
lr=0.1
45/100 Test set: Average loss: 46.9192/9.5062, Accuracy: 14593/2732 (29.19%)/(27.32%)
lr=0.1
46/100 Test set: Average loss: 46.9322/9.5151, Accuracy: 14579/2750 (29.16%)/(27.50%)
lr=0.1
47/100 Test set: Average loss: 46.8929/9.5314, Accuracy: 14582/2759 (29.16%)/(27.59%)
lr=0.1
48/100 Test set: Average loss: 46.9198/9.4959, Accuracy: 14586/2805 (29.17%)/(28.05%)
lr=0.1
49/100 Test set: Average loss: 46.9247/9.5072, Accuracy: 14514/2795 (29.03%)/(27.95%)
lr=0.1
```

```
50/100 Test set: Average loss: 46.8936/9.5291, Accuracy: 14613/2757 (29.23%)/(27.57%)
lr=0.1
51/100 Test set: Average loss: 46.9145/9.5025, Accuracy: 14647/2786 (29.29%)/(27.86%)
lr=0.1
52/100 Test set: Average loss: 46.8786/9.5030, Accuracy: 14764/2801 (29.53%)/(28.01%)
lr=0.1
53/100 Test set: Average loss: 46.9226/9.4896, Accuracy: 14548/2798 (29.10%)/(27.98%)
lr=0.1
54/100 Test set: Average loss: 46.8336/9.4832, Accuracy: 14680/2799 (29.36%)/(27.99%)
lr=0.1
55/100 Test set: Average loss: 46.8355/9.4875, Accuracy: 14638/2796 (29.28%)/(27.96%)
lr=0.1
56/100 Test set: Average loss: 46.8748/9.5346, Accuracy: 14600/2793 (29.20%)/(27.93%)
lr=0.1
57/100 Test set: Average loss: 46.9028/9.5267, Accuracy: 14446/2830 (28.89%)/(28.30%)
lr=0.1
58/100 Test set: Average loss: 46.8533/9.5076, Accuracy: 14708/2791 (29.42%)/(27.91%)
lr=0.1
59/100 Test set: Average loss: 46.7856/9.5021, Accuracy: 14727/2791 (29.45%)/(27.91%)
lr=0.1
60/100 Test set: Average loss: 46.8311/9.4788, Accuracy: 14716/2854 (29.43%)/(28.54%)
lr=0.1
61/100 Test set: Average loss: 46.8017/9.4986, Accuracy: 14689/2730 (29.38%)/(27.30%)
lr=0.1
62/100 Test set: Average loss: 46.7800/9.4989, Accuracy: 14759/2816 (29.52%)/(28.16%)
lr=0.1
63/100 Test set: Average loss: 46.8460/9.5103, Accuracy: 14656/2753 (29.31%)/(27.53%)
lr=0.1
64/100 Test set: Average loss: 46.8224/9.4843, Accuracy: 14688/2821 (29.38%)/(28.21%)
lr=0.1
65/100 Test set: Average loss: 46.7488/9.4920, Accuracy: 14874/2866 (29.75%)/(28.66%)
lr=0.1
66/100 Test set: Average loss: 46.7640/9.4933, Accuracy: 14706/2782 (29.41%)/(27.82%)
lr=0.1
67/100 Test set: Average loss: 46.7765/9.4962, Accuracy: 14757/2826 (29.51%)/(28.26%)
lr=0.1
68/100 Test set: Average loss: 46.7828/9.5018, Accuracy: 14784/2844 (29.57%)/(28.44%)
lr=0.1
69/100 Test set: Average loss: 46.7542/9.5327, Accuracy: 14802/2754 (29.60%)/(27.54%)
lr=0.1
70/100 Test set: Average loss: 46.7485/9.5367, Accuracy: 14833/2825 (29.67%)/(28.25%)
lr=0.1
71/100 Test set: Average loss: 46.7680/9.5147, Accuracy: 14751/2769 (29.50%)/(27.69%)
lr=0.1
72/100 Test set: Average loss: 46.7693/9.5167, Accuracy: 14668/2843 (29.34%)/(28.43%)
lr=0.1
73/100 Test set: Average loss: 46.8265/9.5321, Accuracy: 14787/2824 (29.57%)/(28.24%)
lr=0.1
74/100 Test set: Average loss: 46.7853/9.5062, Accuracy: 14654/2738 (29.31%)/(27.38%)
lr=0.1
75/100 Test set: Average loss: 46.7403/9.4945, Accuracy: 14635/2848 (29.27%)/(28.48%)
lr=0.1
76/100 Test set: Average loss: 46.7396/9.5317, Accuracy: 14782/2767 (29.56%)/(27.67%)
lr=0.1
77/100 Test set: Average loss: 46.7915/9.5072, Accuracy: 14798/2828 (29.60%)/(28.28%)
lr=0.1
78/100 Test set: Average loss: 46.6971/9.5147, Accuracy: 14879/2750 (29.76%)/(27.50%)
lr=0.1
79/100 Test set: Average loss: 46.6788/9.5519, Accuracy: 14840/2809 (29.68%)/(28.09%)
lr=0.1
```

```
80/100 Test set: Average loss: 46.7524/9.5111, Accuracy: 14779/2774 (29.56%)/(27.74%)
lr=0.1
81/100 Test set: Average loss: 46.7096/9.5028, Accuracy: 14748/2795 (29.50%)/(27.95%)
lr=0.1
82/100 Test set: Average loss: 46.7663/9.5049, Accuracy: 14805/2784 (29.61%)/(27.84%)
lr=0.1
83/100 Test set: Average loss: 46.6889/9.5092, Accuracy: 14824/2831 (29.65%)/(28.31%)
lr=0.1
84/100 Test set: Average loss: 46.6716/9.5045, Accuracy: 14899/2782 (29.80%)/(27.82%)
lr=0.1
85/100 Test set: Average loss: 46.7112/9.5407, Accuracy: 14717/2787 (29.43%)/(27.87%)
lr=0.1
86/100 Test set: Average loss: 46.7076/9.5003, Accuracy: 14789/2838 (29.58%)/(28.38%)
lr=0.1
87/100 Test set: Average loss: 46.6940/9.5658, Accuracy: 14878/2757 (29.76%)/(27.57%)
lr=0.1
88/100 Test set: Average loss: 46.6909/9.4929, Accuracy: 14711/2828 (29.42%)/(28.28%)
lr=0.1
89/100 Test set: Average loss: 46.6443/9.5339, Accuracy: 14768/2820 (29.54%)/(28.20%)
lr=0.1
90/100 Test set: Average loss: 46.6865/9.5259, Accuracy: 14916/2799 (29.83%)/(27.99%)
lr=0.1
91/100 Test set: Average loss: 46.6882/9.5138, Accuracy: 14793/2790 (29.59%)/(27.90%)
lr=0.1
92/100 Test set: Average loss: 46.7045/9.5294, Accuracy: 14851/2798 (29.70%)/(27.98%)
lr=0.1
93/100 Test set: Average loss: 46.7076/9.5007, Accuracy: 14676/2820 (29.35%)/(28.20%)
lr=0.1
94/100 Test set: Average loss: 46.6280/9.5346, Accuracy: 14809/2799 (29.62%)/(27.99%)
lr=0.1
95/100 Test set: Average loss: 46.6877/9.5011, Accuracy: 14925/2862 (29.85%)/(28.62%)
lr=0.1
96/100 Test set: Average loss: 46.6248/9.5159, Accuracy: 14854/2751 (29.71%)/(27.51%)
lr=0.1
97/100 Test set: Average loss: 46.6314/9.5449, Accuracy: 14714/2869 (29.43%)/(28.69%)
lr=0.1
98/100 Test set: Average loss: 46.6067/9.5025, Accuracy: 14933/2791 (29.87%)/(27.91%)
lr=0.1
99/100 Test set: Average loss: 46.6137/9.4946, Accuracy: 14822/2852 (29.64%)/(28.52%)
lr=0.1
Files already downloaded and verified
Files already downloaded and verified
['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'tr
uck']
(50000, 32, 32, 3)
(10000, 32, 32, 3)
DNN_MNIST_N(
  (layer1): Sequential(
    (0): Linear(in_features=3072, out_features=3, bias=True)
    (1): BatchNorm1d(3, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tru
e)
    (2): ReLU(inplace=True)
  )
  (layer2): Sequential(
    (0): Linear(in_features=3, out_features=6, bias=True)
    (1): BatchNorm1d(6, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tru
e)
    (2): ReLU(inplace=True)
  )
  (layer3): Sequential(
```

```
    (0): Linear(in_features=6, out_features=3, bias=True)
    (1): BatchNorm1d(3, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tru
e)
    (2): ReLU(inplace=True)
  )
  (layer4): Sequential(
    (0): Linear(in_features=3, out_features=10, bias=True)
  )
)
Number of total parameters:  9328
0/100 Test set: Average loss: 55.7635/10.7256, Accuracy: 8023/1649 (16.05%)/(16.49%)
lr=0.1
1/100 Test set: Average loss: 51.9999/10.2856, Accuracy: 9577/2112 (19.15%)/(21.12%)
lr=0.1
2/100 Test set: Average loss: 50.8975/10.1631, Accuracy: 10665/2109 (21.33%)/(21.09%)
lr=0.1
3/100 Test set: Average loss: 50.3214/10.0254, Accuracy: 11038/2223 (22.08%)/(22.23%)
lr=0.1
4/100 Test set: Average loss: 49.9003/9.9426, Accuracy: 11409/2361 (22.82%)/(23.61%)
lr=0.1
5/100 Test set: Average loss: 49.5390/9.8747, Accuracy: 11856/2435 (23.71%)/(24.35%)
lr=0.1
6/100 Test set: Average loss: 49.1779/9.8095, Accuracy: 12307/2513 (24.61%)/(25.13%)
lr=0.1
7/100 Test set: Average loss: 48.7224/9.7127, Accuracy: 12846/2640 (25.69%)/(26.40%)
lr=0.1
8/100 Test set: Average loss: 48.1881/9.6429, Accuracy: 13368/2668 (26.74%)/(26.68%)
lr=0.1
9/100 Test set: Average loss: 47.7936/9.5821, Accuracy: 13541/2777 (27.08%)/(27.77%)
lr=0.1
10/100 Test set: Average loss: 47.4393/9.5034, Accuracy: 13899/2835 (27.80%)/(28.35%)
lr=0.1
11/100 Test set: Average loss: 47.1473/9.4464, Accuracy: 14318/2951 (28.64%)/(29.51%)
lr=0.1
12/100 Test set: Average loss: 46.7964/9.4074, Accuracy: 14774/3005 (29.55%)/(30.05%)
lr=0.1
13/100 Test set: Average loss: 46.4486/9.3388, Accuracy: 15226/3001 (30.45%)/(30.01%)
lr=0.1
14/100 Test set: Average loss: 46.1162/9.3063, Accuracy: 15414/2982 (30.83%)/(29.82%)
lr=0.1
15/100 Test set: Average loss: 45.9570/9.2335, Accuracy: 15486/3037 (30.97%)/(30.37%)
lr=0.1
16/100 Test set: Average loss: 45.5889/9.1615, Accuracy: 16088/3073 (32.18%)/(30.73%)
lr=0.1
17/100 Test set: Average loss: 45.3801/9.1274, Accuracy: 16290/3143 (32.58%)/(31.43%)
lr=0.1
18/100 Test set: Average loss: 45.2021/9.1781, Accuracy: 16525/3066 (33.05%)/(30.66%)
lr=0.1
19/100 Test set: Average loss: 45.1636/9.1395, Accuracy: 16523/3067 (33.05%)/(30.67%)
lr=0.1
20/100 Test set: Average loss: 44.9831/9.0836, Accuracy: 16798/3207 (33.60%)/(32.07%)
lr=0.1
21/100 Test set: Average loss: 44.8627/9.0839, Accuracy: 16939/3211 (33.88%)/(32.11%)
lr=0.1
22/100 Test set: Average loss: 44.8565/9.0950, Accuracy: 16901/3224 (33.80%)/(32.24%)
lr=0.1
23/100 Test set: Average loss: 44.8287/9.0842, Accuracy: 16934/3223 (33.87%)/(32.23%)
lr=0.1
24/100 Test set: Average loss: 44.7395/9.0703, Accuracy: 17090/3270 (34.18%)/(32.70%)
lr=0.1
```

```
25/100 Test set: Average loss: 44.6756/9.0909, Accuracy: 17103/3248 (34.21%)/(32.48%)
lr=0.1
26/100 Test set: Average loss: 44.6932/9.0427, Accuracy: 17004/3307 (34.01%)/(33.07%)
lr=0.1
27/100 Test set: Average loss: 44.5944/9.1090, Accuracy: 17036/3286 (34.07%)/(32.86%)
lr=0.1
28/100 Test set: Average loss: 44.5906/9.0475, Accuracy: 17192/3300 (34.38%)/(33.00%)
lr=0.1
29/100 Test set: Average loss: 44.5224/9.0226, Accuracy: 17301/3352 (34.60%)/(33.52%)
lr=0.1
30/100 Test set: Average loss: 44.4796/9.0279, Accuracy: 17141/3318 (34.28%)/(33.18%)
lr=0.1
31/100 Test set: Average loss: 44.4014/9.0617, Accuracy: 17227/3291 (34.45%)/(32.91%)
lr=0.1
32/100 Test set: Average loss: 44.3716/9.0332, Accuracy: 17282/3322 (34.56%)/(33.22%)
lr=0.1
33/100 Test set: Average loss: 44.3547/9.0347, Accuracy: 17212/3346 (34.42%)/(33.46%)
lr=0.1
34/100 Test set: Average loss: 44.3369/9.0597, Accuracy: 17255/3334 (34.51%)/(33.34%)
lr=0.1
35/100 Test set: Average loss: 44.3254/9.0460, Accuracy: 17249/3254 (34.50%)/(32.54%)
lr=0.1
36/100 Test set: Average loss: 44.2787/9.0257, Accuracy: 17317/3356 (34.63%)/(33.56%)
lr=0.1
37/100 Test set: Average loss: 44.2276/9.0460, Accuracy: 17322/3303 (34.64%)/(33.03%)
lr=0.1
38/100 Test set: Average loss: 44.3077/9.0553, Accuracy: 17320/3328 (34.64%)/(33.28%)
lr=0.1
39/100 Test set: Average loss: 44.3127/9.0677, Accuracy: 17279/3324 (34.56%)/(33.24%)
lr=0.1
40/100 Test set: Average loss: 44.1692/9.0600, Accuracy: 17387/3306 (34.77%)/(33.06%)
lr=0.1
41/100 Test set: Average loss: 44.2409/9.0536, Accuracy: 17314/3321 (34.63%)/(33.21%)
lr=0.1
42/100 Test set: Average loss: 44.2405/9.0555, Accuracy: 17361/3319 (34.72%)/(33.19%)
lr=0.1
43/100 Test set: Average loss: 44.1154/9.0614, Accuracy: 17489/3324 (34.98%)/(33.24%)
lr=0.1
44/100 Test set: Average loss: 44.1574/9.0650, Accuracy: 17351/3292 (34.70%)/(32.92%)
lr=0.1
45/100 Test set: Average loss: 44.2327/9.0453, Accuracy: 17242/3332 (34.48%)/(33.32%)
lr=0.1
46/100 Test set: Average loss: 44.1443/9.0737, Accuracy: 17405/3301 (34.81%)/(33.01%)
lr=0.1
47/100 Test set: Average loss: 44.1229/9.0362, Accuracy: 17430/3312 (34.86%)/(33.12%)
lr=0.1
48/100 Test set: Average loss: 44.0971/9.0599, Accuracy: 17253/3269 (34.51%)/(32.69%)
lr=0.1
49/100 Test set: Average loss: 44.0658/9.0502, Accuracy: 17365/3327 (34.73%)/(33.27%)
lr=0.1
50/100 Test set: Average loss: 44.1101/9.0483, Accuracy: 17430/3285 (34.86%)/(32.85%)
lr=0.1
51/100 Test set: Average loss: 44.0988/9.0261, Accuracy: 17427/3354 (34.85%)/(33.54%)
lr=0.1
52/100 Test set: Average loss: 44.1028/9.1007, Accuracy: 17450/3295 (34.90%)/(32.95%)
lr=0.1
53/100 Test set: Average loss: 44.0626/9.0516, Accuracy: 17413/3387 (34.83%)/(33.87%)
lr=0.1
54/100 Test set: Average loss: 44.0935/9.0407, Accuracy: 17405/3329 (34.81%)/(33.29%)
lr=0.1
```

```
55/100 Test set: Average loss: 44.0053/9.0392, Accuracy: 17423/3356 (34.85%)/(33.56%)
lr=0.1
56/100 Test set: Average loss: 43.9988/9.1736, Accuracy: 17476/3203 (34.95%)/(32.03%)
lr=0.1
57/100 Test set: Average loss: 44.0073/9.0587, Accuracy: 17469/3316 (34.94%)/(33.16%)
lr=0.1
58/100 Test set: Average loss: 43.9834/9.0382, Accuracy: 17495/3355 (34.99%)/(33.55%)
lr=0.1
59/100 Test set: Average loss: 43.9830/9.0591, Accuracy: 17462/3313 (34.92%)/(33.13%)
lr=0.1
60/100 Test set: Average loss: 43.9460/9.0818, Accuracy: 17554/3244 (35.11%)/(32.44%)
lr=0.1
61/100 Test set: Average loss: 43.9864/9.0799, Accuracy: 17319/3304 (34.64%)/(33.04%)
lr=0.1
62/100 Test set: Average loss: 43.9721/9.0776, Accuracy: 17517/3309 (35.03%)/(33.09%)
lr=0.1
63/100 Test set: Average loss: 43.9285/9.0786, Accuracy: 17523/3290 (35.05%)/(32.90%)
lr=0.1
64/100 Test set: Average loss: 43.8926/9.0485, Accuracy: 17541/3329 (35.08%)/(33.29%)
lr=0.1
65/100 Test set: Average loss: 43.8593/9.0889, Accuracy: 17505/3307 (35.01%)/(33.07%)
lr=0.1
66/100 Test set: Average loss: 43.9197/9.0647, Accuracy: 17504/3234 (35.01%)/(32.34%)
lr=0.1
67/100 Test set: Average loss: 43.9697/9.0610, Accuracy: 17295/3321 (34.59%)/(33.21%)
lr=0.1
68/100 Test set: Average loss: 43.9026/9.0731, Accuracy: 17391/3325 (34.78%)/(33.25%)
lr=0.1
69/100 Test set: Average loss: 43.8807/9.0576, Accuracy: 17515/3290 (35.03%)/(32.90%)
lr=0.1
70/100 Test set: Average loss: 43.8163/9.0551, Accuracy: 17572/3343 (35.14%)/(33.43%)
lr=0.1
71/100 Test set: Average loss: 43.8754/9.0994, Accuracy: 17472/3263 (34.94%)/(32.63%)
lr=0.1
72/100 Test set: Average loss: 43.8071/9.0397, Accuracy: 17485/3339 (34.97%)/(33.39%)
lr=0.1
73/100 Test set: Average loss: 43.7892/9.0751, Accuracy: 17611/3337 (35.22%)/(33.37%)
lr=0.1
74/100 Test set: Average loss: 43.8331/9.0622, Accuracy: 17650/3303 (35.30%)/(33.03%)
lr=0.1
75/100 Test set: Average loss: 43.8288/9.0750, Accuracy: 17519/3281 (35.04%)/(32.81%)
lr=0.1
76/100 Test set: Average loss: 43.7837/9.0566, Accuracy: 17589/3324 (35.18%)/(33.24%)
lr=0.1
77/100 Test set: Average loss: 43.8200/9.0543, Accuracy: 17565/3291 (35.13%)/(32.91%)
lr=0.1
78/100 Test set: Average loss: 43.8026/9.0987, Accuracy: 17520/3317 (35.04%)/(33.17%)
lr=0.1
79/100 Test set: Average loss: 43.8190/9.0760, Accuracy: 17625/3316 (35.25%)/(33.16%)
lr=0.1
80/100 Test set: Average loss: 43.8108/9.0784, Accuracy: 17593/3334 (35.19%)/(33.34%)
lr=0.1
81/100 Test set: Average loss: 43.9009/9.0714, Accuracy: 17573/3258 (35.15%)/(32.58%)
lr=0.1
82/100 Test set: Average loss: 43.7914/9.0631, Accuracy: 17630/3300 (35.26%)/(33.00%)
lr=0.1
83/100 Test set: Average loss: 43.7594/9.0389, Accuracy: 17683/3330 (35.37%)/(33.30%)
lr=0.1
84/100 Test set: Average loss: 43.7759/9.0806, Accuracy: 17686/3340 (35.37%)/(33.40%)
lr=0.1
```

```
85/100 Test set: Average loss: 43.8498/9.0402, Accuracy: 17507/3323 (35.01%)/(33.23%)
lr=0.1
86/100 Test set: Average loss: 43.8010/9.1050, Accuracy: 17605/3284 (35.21%)/(32.84%)
lr=0.1
87/100 Test set: Average loss: 43.8055/9.0503, Accuracy: 17550/3306 (35.10%)/(33.06%)
lr=0.1
88/100 Test set: Average loss: 43.9004/9.0982, Accuracy: 17493/3327 (34.99%)/(33.27%)
lr=0.1
89/100 Test set: Average loss: 43.8403/9.0661, Accuracy: 17538/3348 (35.08%)/(33.48%)
lr=0.1
90/100 Test set: Average loss: 43.8027/9.0576, Accuracy: 17658/3322 (35.32%)/(33.22%)
lr=0.1
91/100 Test set: Average loss: 43.7358/9.0446, Accuracy: 17685/3332 (35.37%)/(33.32%)
lr=0.1
92/100 Test set: Average loss: 43.7413/9.0664, Accuracy: 17642/3252 (35.28%)/(32.52%)
lr=0.1
93/100 Test set: Average loss: 43.6668/9.0590, Accuracy: 17630/3363 (35.26%)/(33.63%)
lr=0.1
94/100 Test set: Average loss: 43.7256/9.0499, Accuracy: 17639/3297 (35.28%)/(32.97%)
lr=0.1
95/100 Test set: Average loss: 43.7876/9.0617, Accuracy: 17602/3313 (35.20%)/(33.13%)
lr=0.1
96/100 Test set: Average loss: 43.6974/9.0772, Accuracy: 17694/3330 (35.39%)/(33.30%)
lr=0.1
97/100 Test set: Average loss: 43.6222/9.0729, Accuracy: 17730/3326 (35.46%)/(33.26%)
lr=0.1
98/100 Test set: Average loss: 43.7514/9.1074, Accuracy: 17704/3187 (35.41%)/(31.87%)
lr=0.1
99/100 Test set: Average loss: 43.7578/9.0912, Accuracy: 17565/3278 (35.13%)/(32.78%)
lr=0.1
Files already downloaded and verified
Files already downloaded and verified
['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'tr
uck']
(50000, 32, 32, 3)
(10000, 32, 32, 3)
DNN_MNIST_N(
  (layer1): Sequential(
    (0): Linear(in_features=3072, out_features=5, bias=True)
    (1): BatchNorm1d(5, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tru
e)
    (2): ReLU(inplace=True)
  )
  (layer2): Sequential(
    (0): Linear(in_features=5, out_features=10, bias=True)
    (1): BatchNorm1d(10, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tr
ue)
    (2): ReLU(inplace=True)
  )
  (layer3): Sequential(
    (0): Linear(in_features=10, out_features=5, bias=True)
    (1): BatchNorm1d(5, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tru
e)
    (2): ReLU(inplace=True)
  )
  (layer4): Sequential(
    (0): Linear(in_features=5, out_features=10, bias=True)
  )
)
Number of total parameters:  15580
```

```
0/100 Test set: Average loss: 54.2727/10.2799, Accuracy: 8273/2258 (16.55%)/(22.58%)
lr=0.1
1/100 Test set: Average loss: 49.8549/9.7284, Accuracy: 11711/2410 (23.42%)/(24.10%)
lr=0.1
2/100 Test set: Average loss: 48.1099/9.5699, Accuracy: 12344/2504 (24.69%)/(25.04%)
lr=0.1
3/100 Test set: Average loss: 47.3105/9.3936, Accuracy: 13193/2678 (26.39%)/(26.78%)
lr=0.1
4/100 Test set: Average loss: 46.7830/9.3021, Accuracy: 13820/2836 (27.64%)/(28.36%)
lr=0.1
5/100 Test set: Average loss: 46.3943/9.2619, Accuracy: 14233/2883 (28.47%)/(28.83%)
lr=0.1
6/100 Test set: Average loss: 46.1096/9.2066, Accuracy: 14845/3003 (29.69%)/(30.03%)
lr=0.1
7/100 Test set: Average loss: 45.7796/9.1512, Accuracy: 15363/3071 (30.73%)/(30.71%)
lr=0.1
8/100 Test set: Average loss: 45.3988/9.0897, Accuracy: 15993/3156 (31.99%)/(31.56%)
lr=0.1
9/100 Test set: Average loss: 45.0103/9.0817, Accuracy: 16429/3242 (32.86%)/(32.42%)
lr=0.1
10/100 Test set: Average loss: 44.7076/8.9810, Accuracy: 16932/3339 (33.86%)/(33.39%)
lr=0.1
11/100 Test set: Average loss: 44.2692/8.8985, Accuracy: 17534/3457 (35.07%)/(34.57%)
lr=0.1
12/100 Test set: Average loss: 43.6983/8.8150, Accuracy: 17963/3568 (35.93%)/(35.68%)
lr=0.1
13/100 Test set: Average loss: 43.2661/8.7147, Accuracy: 18395/3633 (36.79%)/(36.33%)
lr=0.1
14/100 Test set: Average loss: 42.8746/8.6604, Accuracy: 18624/3677 (37.25%)/(36.77%)
lr=0.1
15/100 Test set: Average loss: 42.6102/8.6843, Accuracy: 18836/3667 (37.67%)/(36.67%)
lr=0.1
16/100 Test set: Average loss: 42.5083/8.6947, Accuracy: 18868/3652 (37.74%)/(36.52%)
lr=0.1
17/100 Test set: Average loss: 42.2823/8.5844, Accuracy: 18997/3688 (37.99%)/(36.88%)
lr=0.1
18/100 Test set: Average loss: 42.0759/8.5855, Accuracy: 19104/3708 (38.21%)/(37.08%)
lr=0.1
19/100 Test set: Average loss: 42.0333/8.5878, Accuracy: 19245/3684 (38.49%)/(36.84%)
lr=0.1
20/100 Test set: Average loss: 41.9594/8.5823, Accuracy: 19266/3697 (38.53%)/(36.97%)
lr=0.1
21/100 Test set: Average loss: 41.8446/8.5799, Accuracy: 19205/3701 (38.41%)/(37.01%)
lr=0.1
22/100 Test set: Average loss: 41.7599/8.5941, Accuracy: 19239/3659 (38.48%)/(36.59%)
lr=0.1
23/100 Test set: Average loss: 41.6174/8.6356, Accuracy: 19355/3681 (38.71%)/(36.81%)
lr=0.1
24/100 Test set: Average loss: 41.6672/8.5549, Accuracy: 19296/3668 (38.59%)/(36.68%)
lr=0.1
25/100 Test set: Average loss: 41.6348/8.6732, Accuracy: 19335/3555 (38.67%)/(35.55%)
lr=0.1
26/100 Test set: Average loss: 41.5878/8.5608, Accuracy: 19295/3747 (38.59%)/(37.47%)
lr=0.1
27/100 Test set: Average loss: 41.5447/8.5744, Accuracy: 19435/3734 (38.87%)/(37.34%)
lr=0.1
28/100 Test set: Average loss: 41.4233/8.5122, Accuracy: 19437/3763 (38.87%)/(37.63%)
lr=0.1
29/100 Test set: Average loss: 41.3358/8.5408, Accuracy: 19467/3704 (38.93%)/(37.04%)
lr=0.1
```

```
30/100 Test set: Average loss: 41.4013/8.5557, Accuracy: 19498/3714 (39.00%)/(37.14%)
lr=0.1
31/100 Test set: Average loss: 41.3690/8.5911, Accuracy: 19451/3674 (38.90%)/(36.74%)
lr=0.1
32/100 Test set: Average loss: 41.3611/8.5342, Accuracy: 19528/3678 (39.06%)/(36.78%)
lr=0.1
33/100 Test set: Average loss: 41.3266/8.5403, Accuracy: 19469/3707 (38.94%)/(37.07%)
lr=0.1
34/100 Test set: Average loss: 41.2319/8.5780, Accuracy: 19604/3696 (39.21%)/(36.96%)
lr=0.1
35/100 Test set: Average loss: 41.2493/8.5453, Accuracy: 19470/3742 (38.94%)/(37.42%)
lr=0.1
36/100 Test set: Average loss: 41.1492/8.5224, Accuracy: 19626/3741 (39.25%)/(37.41%)
lr=0.1
37/100 Test set: Average loss: 41.1085/8.5329, Accuracy: 19674/3711 (39.35%)/(37.11%)
lr=0.1
38/100 Test set: Average loss: 41.0809/8.5526, Accuracy: 19765/3664 (39.53%)/(36.64%)
lr=0.1
39/100 Test set: Average loss: 41.0718/8.5580, Accuracy: 19719/3727 (39.44%)/(37.27%)
lr=0.1
40/100 Test set: Average loss: 41.0342/8.5950, Accuracy: 19601/3674 (39.20%)/(36.74%)
lr=0.1
41/100 Test set: Average loss: 41.0463/8.5588, Accuracy: 19697/3740 (39.39%)/(37.40%)
lr=0.1
42/100 Test set: Average loss: 41.0696/8.5533, Accuracy: 19682/3696 (39.36%)/(36.96%)
lr=0.1
43/100 Test set: Average loss: 40.9335/8.5418, Accuracy: 19722/3704 (39.44%)/(37.04%)
lr=0.1
44/100 Test set: Average loss: 40.9701/8.5566, Accuracy: 19850/3691 (39.70%)/(36.91%)
lr=0.1
45/100 Test set: Average loss: 40.8976/8.5339, Accuracy: 19790/3674 (39.58%)/(36.74%)
lr=0.1
46/100 Test set: Average loss: 40.8202/8.5562, Accuracy: 19917/3741 (39.83%)/(37.41%)
lr=0.1
47/100 Test set: Average loss: 40.9120/8.6210, Accuracy: 19731/3696 (39.46%)/(36.96%)
lr=0.1
48/100 Test set: Average loss: 40.9092/8.5809, Accuracy: 19713/3634 (39.43%)/(36.34%)
lr=0.1
49/100 Test set: Average loss: 40.9953/8.5607, Accuracy: 19707/3723 (39.41%)/(37.23%)
lr=0.1
50/100 Test set: Average loss: 40.7779/8.5420, Accuracy: 19912/3716 (39.82%)/(37.16%)
lr=0.1
51/100 Test set: Average loss: 40.9060/8.5716, Accuracy: 19808/3698 (39.62%)/(36.98%)
lr=0.1
52/100 Test set: Average loss: 40.8920/8.5990, Accuracy: 19824/3676 (39.65%)/(36.76%)
lr=0.1
53/100 Test set: Average loss: 40.7985/8.5357, Accuracy: 19844/3717 (39.69%)/(37.17%)
lr=0.1
54/100 Test set: Average loss: 40.6991/8.5416, Accuracy: 19861/3719 (39.72%)/(37.19%)
lr=0.1
55/100 Test set: Average loss: 40.7105/8.5289, Accuracy: 20051/3759 (40.10%)/(37.59%)
lr=0.1
56/100 Test set: Average loss: 40.7409/8.5691, Accuracy: 19986/3724 (39.97%)/(37.24%)
lr=0.1
57/100 Test set: Average loss: 40.6508/8.5541, Accuracy: 19927/3687 (39.85%)/(36.87%)
lr=0.1
58/100 Test set: Average loss: 40.6485/8.5744, Accuracy: 19944/3674 (39.89%)/(36.74%)
lr=0.1
59/100 Test set: Average loss: 40.7280/8.5763, Accuracy: 19871/3680 (39.74%)/(36.80%)
lr=0.1
```

```
60/100 Test set: Average loss: 40.5585/8.5691, Accuracy: 20023/3667 (40.05%)/(36.67%)
lr=0.1
61/100 Test set: Average loss: 40.6246/8.5992, Accuracy: 20023/3668 (40.05%)/(36.68%)
lr=0.1
62/100 Test set: Average loss: 40.5508/8.6120, Accuracy: 20041/3661 (40.08%)/(36.61%)
lr=0.1
63/100 Test set: Average loss: 40.5547/8.5946, Accuracy: 19972/3673 (39.94%)/(36.73%)
lr=0.1
64/100 Test set: Average loss: 40.5904/8.5884, Accuracy: 19956/3714 (39.91%)/(37.14%)
lr=0.1
65/100 Test set: Average loss: 40.5238/8.5904, Accuracy: 20042/3703 (40.08%)/(37.03%)
lr=0.1
66/100 Test set: Average loss: 40.5198/8.5780, Accuracy: 20156/3711 (40.31%)/(37.11%)
lr=0.1
67/100 Test set: Average loss: 40.6106/8.6027, Accuracy: 19899/3709 (39.80%)/(37.09%)
lr=0.1
68/100 Test set: Average loss: 40.5989/8.6128, Accuracy: 20040/3717 (40.08%)/(37.17%)
lr=0.1
69/100 Test set: Average loss: 40.5860/8.6163, Accuracy: 20119/3617 (40.24%)/(36.17%)
lr=0.1
70/100 Test set: Average loss: 40.5251/8.5985, Accuracy: 20013/3683 (40.03%)/(36.83%)
lr=0.1
71/100 Test set: Average loss: 40.4220/8.6194, Accuracy: 20153/3716 (40.31%)/(37.16%)
lr=0.1
72/100 Test set: Average loss: 40.6101/8.5694, Accuracy: 20016/3696 (40.03%)/(36.96%)
lr=0.1
73/100 Test set: Average loss: 40.5125/8.6483, Accuracy: 20151/3680 (40.30%)/(36.80%)
lr=0.1
74/100 Test set: Average loss: 40.5121/8.6229, Accuracy: 20088/3700 (40.18%)/(37.00%)
lr=0.1
75/100 Test set: Average loss: 40.4112/8.5596, Accuracy: 20147/3703 (40.29%)/(37.03%)
lr=0.1
76/100 Test set: Average loss: 40.4172/8.6140, Accuracy: 20134/3716 (40.27%)/(37.16%)
lr=0.1
77/100 Test set: Average loss: 40.5780/8.6085, Accuracy: 19962/3731 (39.92%)/(37.31%)
lr=0.1
78/100 Test set: Average loss: 40.3599/8.6037, Accuracy: 20241/3671 (40.48%)/(36.71%)
lr=0.1
79/100 Test set: Average loss: 40.4012/8.6229, Accuracy: 20213/3733 (40.43%)/(37.33%)
lr=0.1
80/100 Test set: Average loss: 40.4408/8.6453, Accuracy: 20112/3688 (40.22%)/(36.88%)
lr=0.1
81/100 Test set: Average loss: 40.3797/8.5800, Accuracy: 20188/3706 (40.38%)/(37.06%)
lr=0.1
82/100 Test set: Average loss: 40.3505/8.6823, Accuracy: 20140/3705 (40.28%)/(37.05%)
lr=0.1
83/100 Test set: Average loss: 40.4077/8.5910, Accuracy: 20111/3715 (40.22%)/(37.15%)
lr=0.1
84/100 Test set: Average loss: 40.3624/8.6017, Accuracy: 20173/3661 (40.35%)/(36.61%)
lr=0.1
85/100 Test set: Average loss: 40.3044/8.6159, Accuracy: 20201/3707 (40.40%)/(37.07%)
lr=0.1
86/100 Test set: Average loss: 40.3413/8.6383, Accuracy: 20215/3694 (40.43%)/(36.94%)
lr=0.1
87/100 Test set: Average loss: 40.4064/8.6568, Accuracy: 20116/3745 (40.23%)/(37.45%)
lr=0.1
88/100 Test set: Average loss: 40.3613/8.6174, Accuracy: 20186/3695 (40.37%)/(36.95%)
lr=0.1
89/100 Test set: Average loss: 40.3357/8.5942, Accuracy: 20175/3692 (40.35%)/(36.92%)
lr=0.1
```

```
90/100 Test set: Average loss: 40.2913/8.5798, Accuracy: 20285/3680 (40.57%)/(36.80%)
lr=0.1
91/100 Test set: Average loss: 40.3025/8.5997, Accuracy: 20261/3703 (40.52%)/(37.03%)
lr=0.1
92/100 Test set: Average loss: 40.3357/8.6176, Accuracy: 20230/3678 (40.46%)/(36.78%)
lr=0.1
93/100 Test set: Average loss: 40.1715/8.5965, Accuracy: 20248/3692 (40.50%)/(36.92%)
lr=0.1
94/100 Test set: Average loss: 40.2323/8.6260, Accuracy: 20303/3672 (40.61%)/(36.72%)
lr=0.1
95/100 Test set: Average loss: 40.2875/8.6052, Accuracy: 20226/3695 (40.45%)/(36.95%)
lr=0.1
96/100 Test set: Average loss: 40.2996/8.6644, Accuracy: 20152/3650 (40.30%)/(36.50%)
lr=0.1
97/100 Test set: Average loss: 40.3142/8.6245, Accuracy: 20144/3691 (40.29%)/(36.91%)
lr=0.1
98/100 Test set: Average loss: 40.2028/8.6284, Accuracy: 20286/3667 (40.57%)/(36.67%)
lr=0.1
99/100 Test set: Average loss: 40.2315/8.6226, Accuracy: 20238/3698 (40.48%)/(36.98%)
lr=0.1
Files already downloaded and verified
Files already downloaded and verified
['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'tr
uck']
(50000, 32, 32, 3)
(10000, 32, 32, 3)
DNN_MNIST_N(
  (layer1): Sequential(
    (0): Linear(in_features=3072, out_features=10, bias=True)
    (1): BatchNorm1d(10, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tr
ue)
    (2): ReLU(inplace=True)
  )
  (layer2): Sequential(
    (0): Linear(in_features=10, out_features=20, bias=True)
    (1): BatchNorm1d(20, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tr
ue)
    (2): ReLU(inplace=True)
  )
  (layer3): Sequential(
    (0): Linear(in_features=20, out_features=10, bias=True)
    (1): BatchNorm1d(10, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tr
ue)
    (2): ReLU(inplace=True)
  )
  (layer4): Sequential(
    (0): Linear(in_features=10, out_features=10, bias=True)
  )
)
Number of total parameters:  31350
0/100 Test set: Average loss: 53.0012/9.7758, Accuracy: 11191/2949 (22.38%)/(29.49%)
lr=0.1
1/100 Test set: Average loss: 46.6267/8.9460, Accuracy: 16043/3504 (32.09%)/(35.04%)
lr=0.1
2/100 Test set: Average loss: 43.5760/8.5312, Accuracy: 18290/3833 (36.58%)/(38.33%)
lr=0.1
3/100 Test set: Average loss: 41.9932/8.3351, Accuracy: 19682/3983 (39.36%)/(39.83%)
lr=0.1
4/100 Test set: Average loss: 41.1636/8.2434, Accuracy: 20394/4124 (40.79%)/(41.24%)
lr=0.1
```

```
5/100 Test set: Average loss: 40.4855/8.1946, Accuracy: 20959/4135 (41.92%)/(41.35%)
lr=0.1
6/100 Test set: Average loss: 40.0230/8.0574, Accuracy: 21375/4262 (42.75%)/(42.62%)
lr=0.1
7/100 Test set: Average loss: 39.5908/7.9830, Accuracy: 21678/4303 (43.36%)/(43.03%)
lr=0.1
8/100 Test set: Average loss: 39.0701/7.9954, Accuracy: 22056/4265 (44.11%)/(42.65%)
lr=0.1
9/100 Test set: Average loss: 38.7551/7.9993, Accuracy: 22311/4307 (44.62%)/(43.07%)
lr=0.1
10/100 Test set: Average loss: 38.4390/7.9284, Accuracy: 22436/4341 (44.87%)/(43.41%)
lr=0.1
11/100 Test set: Average loss: 38.3501/7.8679, Accuracy: 22595/4429 (45.19%)/(44.29%)
lr=0.1
12/100 Test set: Average loss: 38.1617/7.9023, Accuracy: 22712/4362 (45.42%)/(43.62%)
lr=0.1
13/100 Test set: Average loss: 37.9197/7.9308, Accuracy: 22914/4376 (45.83%)/(43.76%)
lr=0.1
14/100 Test set: Average loss: 37.7507/7.9183, Accuracy: 23082/4361 (46.16%)/(43.61%)
lr=0.1
15/100 Test set: Average loss: 37.6879/7.8707, Accuracy: 23035/4410 (46.07%)/(44.10%)
lr=0.1
16/100 Test set: Average loss: 37.4636/7.8785, Accuracy: 23246/4396 (46.49%)/(43.96%)
lr=0.1
17/100 Test set: Average loss: 37.3844/7.8893, Accuracy: 23379/4383 (46.76%)/(43.83%)
lr=0.1
18/100 Test set: Average loss: 37.2216/7.8088, Accuracy: 23488/4443 (46.98%)/(44.43%)
lr=0.1
19/100 Test set: Average loss: 37.2042/7.8638, Accuracy: 23426/4428 (46.85%)/(44.28%)
lr=0.1
20/100 Test set: Average loss: 37.0880/7.8579, Accuracy: 23491/4423 (46.98%)/(44.23%)
lr=0.1
21/100 Test set: Average loss: 36.8908/7.8886, Accuracy: 23727/4350 (47.45%)/(43.50%)
lr=0.1
22/100 Test set: Average loss: 36.7832/7.8223, Accuracy: 23725/4454 (47.45%)/(44.54%)
lr=0.1
23/100 Test set: Average loss: 36.6961/7.8892, Accuracy: 23872/4361 (47.74%)/(43.61%)
lr=0.1
24/100 Test set: Average loss: 36.6223/7.8531, Accuracy: 23877/4452 (47.75%)/(44.52%)
lr=0.1
25/100 Test set: Average loss: 36.4678/7.8612, Accuracy: 24042/4418 (48.08%)/(44.18%)
lr=0.1
26/100 Test set: Average loss: 36.4504/7.8061, Accuracy: 23976/4451 (47.95%)/(44.51%)
lr=0.1
27/100 Test set: Average loss: 36.2749/7.8441, Accuracy: 24129/4474 (48.26%)/(44.74%)
lr=0.1
28/100 Test set: Average loss: 36.2293/7.8405, Accuracy: 24138/4448 (48.28%)/(44.48%)
lr=0.1
29/100 Test set: Average loss: 36.1080/7.7952, Accuracy: 24303/4500 (48.61%)/(45.00%)
lr=0.1
30/100 Test set: Average loss: 35.9617/7.8069, Accuracy: 24323/4487 (48.65%)/(44.87%)
lr=0.1
31/100 Test set: Average loss: 36.0635/7.7530, Accuracy: 24251/4517 (48.50%)/(45.17%)
lr=0.1
32/100 Test set: Average loss: 36.0919/7.8157, Accuracy: 24228/4446 (48.46%)/(44.46%)
lr=0.1
33/100 Test set: Average loss: 35.9530/7.8045, Accuracy: 24277/4498 (48.55%)/(44.98%)
lr=0.1
34/100 Test set: Average loss: 35.7974/7.8324, Accuracy: 24430/4436 (48.86%)/(44.36%)
lr=0.1
```

```
35/100 Test set: Average loss: 35.6967/7.7960, Accuracy: 24573/4483 (49.15%)/(44.83%)
lr=0.1
36/100 Test set: Average loss: 35.5413/7.7931, Accuracy: 24630/4487 (49.26%)/(44.87%)
lr=0.1
37/100 Test set: Average loss: 35.7355/7.8109, Accuracy: 24481/4463 (48.96%)/(44.63%)
lr=0.1
38/100 Test set: Average loss: 35.5179/7.8159, Accuracy: 24614/4471 (49.23%)/(44.71%)
lr=0.1
39/100 Test set: Average loss: 35.5603/7.7912, Accuracy: 24639/4476 (49.28%)/(44.76%)
lr=0.1
40/100 Test set: Average loss: 35.4822/7.8015, Accuracy: 24601/4420 (49.20%)/(44.20%)
lr=0.1
41/100 Test set: Average loss: 35.3687/7.8192, Accuracy: 24754/4439 (49.51%)/(44.39%)
lr=0.1
42/100 Test set: Average loss: 35.4682/7.7820, Accuracy: 24680/4517 (49.36%)/(45.17%)
lr=0.1
43/100 Test set: Average loss: 35.2313/7.7621, Accuracy: 24913/4459 (49.83%)/(44.59%)
lr=0.1
44/100 Test set: Average loss: 35.2662/7.8713, Accuracy: 24814/4440 (49.63%)/(44.40%)
lr=0.1
45/100 Test set: Average loss: 35.3186/7.7756, Accuracy: 24767/4417 (49.53%)/(44.17%)
lr=0.1
46/100 Test set: Average loss: 35.1551/7.8176, Accuracy: 24840/4448 (49.68%)/(44.48%)
lr=0.1
47/100 Test set: Average loss: 35.0163/7.7943, Accuracy: 24988/4462 (49.98%)/(44.62%)
lr=0.1
48/100 Test set: Average loss: 35.1521/7.7905, Accuracy: 24864/4446 (49.73%)/(44.46%)
lr=0.1
49/100 Test set: Average loss: 35.0692/7.8049, Accuracy: 24888/4483 (49.78%)/(44.83%)
lr=0.1
50/100 Test set: Average loss: 35.0538/7.8392, Accuracy: 24790/4452 (49.58%)/(44.52%)
lr=0.1
51/100 Test set: Average loss: 34.9829/7.8711, Accuracy: 24992/4419 (49.98%)/(44.19%)
lr=0.1
52/100 Test set: Average loss: 34.9199/7.7972, Accuracy: 25036/4464 (50.07%)/(44.64%)
lr=0.1
53/100 Test set: Average loss: 34.9980/7.7935, Accuracy: 25049/4444 (50.10%)/(44.44%)
lr=0.1
54/100 Test set: Average loss: 34.9117/7.8462, Accuracy: 25004/4465 (50.01%)/(44.65%)
lr=0.1
55/100 Test set: Average loss: 34.9177/7.8399, Accuracy: 24965/4418 (49.93%)/(44.18%)
lr=0.1
56/100 Test set: Average loss: 34.8538/7.8044, Accuracy: 25126/4438 (50.25%)/(44.38%)
lr=0.1
57/100 Test set: Average loss: 34.8032/7.8248, Accuracy: 25080/4459 (50.16%)/(44.59%)
lr=0.1
58/100 Test set: Average loss: 34.8126/7.8491, Accuracy: 25142/4429 (50.28%)/(44.29%)
lr=0.1
59/100 Test set: Average loss: 34.7073/7.8587, Accuracy: 25170/4441 (50.34%)/(44.41%)
lr=0.1
60/100 Test set: Average loss: 34.7554/7.8036, Accuracy: 25130/4455 (50.26%)/(44.55%)
lr=0.1
61/100 Test set: Average loss: 34.6453/7.8158, Accuracy: 25217/4472 (50.43%)/(44.72%)
lr=0.1
62/100 Test set: Average loss: 34.6076/7.8374, Accuracy: 25272/4457 (50.54%)/(44.57%)
lr=0.1
63/100 Test set: Average loss: 34.6366/7.8760, Accuracy: 25235/4455 (50.47%)/(44.55%)
lr=0.1
64/100 Test set: Average loss: 34.5680/7.8437, Accuracy: 25306/4458 (50.61%)/(44.58%)
lr=0.1
```

```
65/100 Test set: Average loss: 34.5538/7.8639, Accuracy: 25244/4437 (50.49%)/(44.37%)
lr=0.1
66/100 Test set: Average loss: 34.5233/7.8342, Accuracy: 25338/4449 (50.68%)/(44.49%)
lr=0.1
67/100 Test set: Average loss: 34.3638/7.9011, Accuracy: 25507/4423 (51.01%)/(44.23%)
lr=0.1
68/100 Test set: Average loss: 34.3726/7.8783, Accuracy: 25428/4477 (50.86%)/(44.77%)
lr=0.1
69/100 Test set: Average loss: 34.3617/7.8528, Accuracy: 25364/4469 (50.73%)/(44.69%)
lr=0.1
70/100 Test set: Average loss: 34.2972/7.8540, Accuracy: 25469/4441 (50.94%)/(44.41%)
lr=0.1
71/100 Test set: Average loss: 34.3406/7.9394, Accuracy: 25392/4359 (50.78%)/(43.59%)
lr=0.1
72/100 Test set: Average loss: 34.3424/7.8713, Accuracy: 25404/4448 (50.81%)/(44.48%)
lr=0.1
73/100 Test set: Average loss: 34.3735/7.9266, Accuracy: 25375/4427 (50.75%)/(44.27%)
lr=0.1
74/100 Test set: Average loss: 34.4091/7.8337, Accuracy: 25419/4445 (50.84%)/(44.45%)
lr=0.1
75/100 Test set: Average loss: 34.2514/7.8426, Accuracy: 25483/4470 (50.97%)/(44.70%)
lr=0.1
76/100 Test set: Average loss: 34.3926/7.8883, Accuracy: 25310/4434 (50.62%)/(44.34%)
lr=0.1
77/100 Test set: Average loss: 34.3900/7.8328, Accuracy: 25466/4447 (50.93%)/(44.47%)
lr=0.1
78/100 Test set: Average loss: 34.1537/7.9109, Accuracy: 25546/4421 (51.09%)/(44.21%)
lr=0.1
79/100 Test set: Average loss: 34.0630/7.8160, Accuracy: 25628/4472 (51.26%)/(44.72%)
lr=0.1
80/100 Test set: Average loss: 34.0188/7.9137, Accuracy: 25665/4424 (51.33%)/(44.24%)
lr=0.1
81/100 Test set: Average loss: 34.2069/7.9816, Accuracy: 25503/4391 (51.01%)/(43.91%)
lr=0.1
82/100 Test set: Average loss: 34.1534/7.8937, Accuracy: 25519/4404 (51.04%)/(44.04%)
lr=0.1
83/100 Test set: Average loss: 34.0186/7.8993, Accuracy: 25675/4407 (51.35%)/(44.07%)
lr=0.1
84/100 Test set: Average loss: 33.9983/7.8706, Accuracy: 25606/4443 (51.21%)/(44.43%)
lr=0.1
85/100 Test set: Average loss: 34.0847/7.9519, Accuracy: 25655/4397 (51.31%)/(43.97%)
lr=0.1
86/100 Test set: Average loss: 34.0603/7.8850, Accuracy: 25706/4428 (51.41%)/(44.28%)
lr=0.1
87/100 Test set: Average loss: 33.8135/7.8972, Accuracy: 25713/4437 (51.43%)/(44.37%)
lr=0.1
88/100 Test set: Average loss: 33.8949/7.9242, Accuracy: 25769/4446 (51.54%)/(44.46%)
lr=0.1
89/100 Test set: Average loss: 33.9336/7.8924, Accuracy: 25689/4422 (51.38%)/(44.22%)
lr=0.1
90/100 Test set: Average loss: 33.8152/7.9616, Accuracy: 25783/4417 (51.57%)/(44.17%)
lr=0.1
91/100 Test set: Average loss: 33.9361/7.9522, Accuracy: 25616/4384 (51.23%)/(43.84%)
lr=0.1
92/100 Test set: Average loss: 33.9056/7.9535, Accuracy: 25777/4369 (51.55%)/(43.69%)
lr=0.1
93/100 Test set: Average loss: 33.8220/7.9814, Accuracy: 25853/4374 (51.71%)/(43.74%)
lr=0.1
94/100 Test set: Average loss: 33.9812/7.9214, Accuracy: 25670/4436 (51.34%)/(44.36%)
lr=0.1
```

```
95/100 Test set: Average loss: 33.8694/7.9160, Accuracy: 25835/4440 (51.67%)/(44.40%)
lr=0.1
96/100 Test set: Average loss: 33.6970/7.9579, Accuracy: 25919/4431 (51.84%)/(44.31%)
lr=0.1
97/100 Test set: Average loss: 33.8064/7.9423, Accuracy: 25839/4427 (51.68%)/(44.27%)
lr=0.1
98/100 Test set: Average loss: 33.6638/7.9381, Accuracy: 25852/4452 (51.70%)/(44.52%)
lr=0.1
99/100 Test set: Average loss: 33.6182/7.9386, Accuracy: 25880/4407 (51.76%)/(44.07%)
lr=0.1
Files already downloaded and verified
Files already downloaded and verified
['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'tr
uck']
(50000, 32, 32, 3)
(10000, 32, 32, 3)
DNN_MNIST_N(
  (layer1): Sequential(
    (0): Linear(in_features=3072, out_features=20, bias=True)
    (1): BatchNorm1d(20, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tr
ue)
    (2): ReLU(inplace=True)
  )
  (layer2): Sequential(
    (0): Linear(in_features=20, out_features=40, bias=True)
    (1): BatchNorm1d(40, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tr
ue)
    (2): ReLU(inplace=True)
  )
  (layer3): Sequential(
    (0): Linear(in_features=40, out_features=20, bias=True)
    (1): BatchNorm1d(20, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tr
ue)
    (2): ReLU(inplace=True)
  )
  (layer4): Sequential(
    (0): Linear(in_features=20, out_features=10, bias=True)
  )
)
Number of total parameters:  63490
0/100 Test set: Average loss: 49.9733/9.0447, Accuracy: 13943/3514 (27.89%)/(35.14%)
lr=0.1
1/100 Test set: Average loss: 43.2918/8.3501, Accuracy: 18810/3975 (37.62%)/(39.75%)
lr=0.1
2/100 Test set: Average loss: 40.8216/8.0453, Accuracy: 20840/4253 (41.68%)/(42.53%)
lr=0.1
3/100 Test set: Average loss: 39.3793/7.8657, Accuracy: 21936/4389 (43.87%)/(43.89%)
lr=0.1
4/100 Test set: Average loss: 38.3244/7.7407, Accuracy: 22684/4472 (45.37%)/(44.72%)
lr=0.1
5/100 Test set: Average loss: 37.5014/7.6606, Accuracy: 23284/4532 (46.57%)/(45.32%)
lr=0.1
6/100 Test set: Average loss: 36.7669/7.5853, Accuracy: 23670/4594 (47.34%)/(45.94%)
lr=0.1
7/100 Test set: Average loss: 36.0770/7.4633, Accuracy: 24174/4655 (48.35%)/(46.55%)
lr=0.1
8/100 Test set: Average loss: 35.5074/7.4802, Accuracy: 24611/4675 (49.22%)/(46.75%)
lr=0.1
9/100 Test set: Average loss: 35.1767/7.4360, Accuracy: 24935/4694 (49.87%)/(46.94%)
lr=0.1
```

```
10/100 Test set: Average loss: 34.8057/7.3727, Accuracy: 25247/4773 (50.49%)/(47.73%)
lr=0.1
11/100 Test set: Average loss: 34.2108/7.3710, Accuracy: 25485/4735 (50.97%)/(47.35%)
lr=0.1
12/100 Test set: Average loss: 33.9408/7.3589, Accuracy: 25866/4803 (51.73%)/(48.03%)
lr=0.1
13/100 Test set: Average loss: 33.6446/7.3509, Accuracy: 26034/4742 (52.07%)/(47.42%)
lr=0.1
14/100 Test set: Average loss: 33.4661/7.3960, Accuracy: 26209/4715 (52.42%)/(47.15%)
lr=0.1
15/100 Test set: Average loss: 33.1773/7.3338, Accuracy: 26254/4748 (52.51%)/(47.48%)
lr=0.1
16/100 Test set: Average loss: 32.8771/7.2883, Accuracy: 26516/4812 (53.03%)/(48.12%)
lr=0.1
17/100 Test set: Average loss: 32.5889/7.3882, Accuracy: 26723/4779 (53.45%)/(47.79%)
lr=0.1
18/100 Test set: Average loss: 32.6585/7.2944, Accuracy: 26669/4825 (53.34%)/(48.25%)
lr=0.1
19/100 Test set: Average loss: 32.2800/7.3324, Accuracy: 27010/4832 (54.02%)/(48.32%)
lr=0.1
20/100 Test set: Average loss: 31.9936/7.3427, Accuracy: 27126/4790 (54.25%)/(47.90%)
lr=0.1
21/100 Test set: Average loss: 31.7792/7.3461, Accuracy: 27340/4845 (54.68%)/(48.45%)
lr=0.1
22/100 Test set: Average loss: 31.8557/7.3305, Accuracy: 27274/4836 (54.55%)/(48.36%)
lr=0.1
23/100 Test set: Average loss: 31.5119/7.3244, Accuracy: 27507/4872 (55.01%)/(48.72%)
lr=0.1
24/100 Test set: Average loss: 31.4591/7.3520, Accuracy: 27475/4795 (54.95%)/(47.95%)
lr=0.1
25/100 Test set: Average loss: 31.4692/7.3074, Accuracy: 27483/4847 (54.97%)/(48.47%)
lr=0.1
26/100 Test set: Average loss: 31.3429/7.3518, Accuracy: 27645/4834 (55.29%)/(48.34%)
lr=0.1
27/100 Test set: Average loss: 31.1739/7.3431, Accuracy: 27777/4861 (55.55%)/(48.61%)
lr=0.1
28/100 Test set: Average loss: 31.0193/7.3722, Accuracy: 27923/4861 (55.85%)/(48.61%)
lr=0.1
29/100 Test set: Average loss: 30.8284/7.4160, Accuracy: 27980/4830 (55.96%)/(48.30%)
lr=0.1
30/100 Test set: Average loss: 30.7777/7.3659, Accuracy: 28059/4843 (56.12%)/(48.43%)
lr=0.1
31/100 Test set: Average loss: 30.5010/7.3473, Accuracy: 28316/4856 (56.63%)/(48.56%)
lr=0.1
32/100 Test set: Average loss: 30.4990/7.4721, Accuracy: 28298/4785 (56.60%)/(47.85%)
lr=0.1
33/100 Test set: Average loss: 30.2766/7.4757, Accuracy: 28418/4796 (56.84%)/(47.96%)
lr=0.1
34/100 Test set: Average loss: 30.3079/7.4008, Accuracy: 28489/4855 (56.98%)/(48.55%)
lr=0.1
35/100 Test set: Average loss: 29.9763/7.3959, Accuracy: 28537/4887 (57.07%)/(48.87%)
lr=0.1
36/100 Test set: Average loss: 29.9327/7.4849, Accuracy: 28686/4863 (57.37%)/(48.63%)
lr=0.1
37/100 Test set: Average loss: 29.8425/7.4736, Accuracy: 28689/4785 (57.38%)/(47.85%)
lr=0.1
38/100 Test set: Average loss: 29.9123/7.5180, Accuracy: 28686/4807 (57.37%)/(48.07%)
lr=0.1
39/100 Test set: Average loss: 29.7858/7.4662, Accuracy: 28656/4842 (57.31%)/(48.42%)
lr=0.1
```

```
40/100 Test set: Average loss: 29.8614/7.5130, Accuracy: 28703/4821 (57.41%)/(48.21%)
lr=0.1
41/100 Test set: Average loss: 29.8808/7.4654, Accuracy: 28688/4848 (57.38%)/(48.48%)
lr=0.1
42/100 Test set: Average loss: 29.8057/7.4817, Accuracy: 28821/4800 (57.64%)/(48.00%)
lr=0.1
43/100 Test set: Average loss: 29.4638/7.4673, Accuracy: 29010/4828 (58.02%)/(48.28%)
lr=0.1
44/100 Test set: Average loss: 29.4154/7.4535, Accuracy: 29027/4883 (58.05%)/(48.83%)
lr=0.1
45/100 Test set: Average loss: 29.2896/7.5176, Accuracy: 29245/4828 (58.49%)/(48.28%)
lr=0.1
46/100 Test set: Average loss: 29.3568/7.5223, Accuracy: 29048/4825 (58.10%)/(48.25%)
lr=0.1
47/100 Test set: Average loss: 29.2231/7.6457, Accuracy: 29180/4800 (58.36%)/(48.00%)
lr=0.1
48/100 Test set: Average loss: 29.1065/7.5631, Accuracy: 29215/4815 (58.43%)/(48.15%)
lr=0.1
49/100 Test set: Average loss: 28.9502/7.5516, Accuracy: 29391/4833 (58.78%)/(48.33%)
lr=0.1
50/100 Test set: Average loss: 29.1297/7.6435, Accuracy: 29232/4774 (58.46%)/(47.74%)
lr=0.1
51/100 Test set: Average loss: 28.8521/7.5014, Accuracy: 29465/4841 (58.93%)/(48.41%)
lr=0.1
52/100 Test set: Average loss: 28.7674/7.6985, Accuracy: 29499/4764 (59.00%)/(47.64%)
lr=0.1
53/100 Test set: Average loss: 28.8593/7.6600, Accuracy: 29427/4796 (58.85%)/(47.96%)
lr=0.1
54/100 Test set: Average loss: 28.7258/7.6990, Accuracy: 29542/4763 (59.08%)/(47.63%)
lr=0.1
55/100 Test set: Average loss: 28.7873/7.6096, Accuracy: 29478/4831 (58.96%)/(48.31%)
lr=0.1
56/100 Test set: Average loss: 28.5274/7.6709, Accuracy: 29668/4772 (59.34%)/(47.72%)
lr=0.1
57/100 Test set: Average loss: 28.4975/7.6683, Accuracy: 29665/4776 (59.33%)/(47.76%)
lr=0.1
58/100 Test set: Average loss: 28.5054/7.6521, Accuracy: 29695/4801 (59.39%)/(48.01%)
lr=0.1
59/100 Test set: Average loss: 28.5829/7.6110, Accuracy: 29567/4842 (59.13%)/(48.42%)
lr=0.1
60/100 Test set: Average loss: 28.2648/7.7161, Accuracy: 29872/4745 (59.74%)/(47.45%)
lr=0.1
61/100 Test set: Average loss: 28.4623/7.6992, Accuracy: 29767/4770 (59.53%)/(47.70%)
lr=0.1
62/100 Test set: Average loss: 28.3862/7.6439, Accuracy: 29834/4838 (59.67%)/(48.38%)
lr=0.1
63/100 Test set: Average loss: 28.1566/7.6896, Accuracy: 29939/4817 (59.88%)/(48.17%)
lr=0.1
64/100 Test set: Average loss: 28.0497/7.7329, Accuracy: 30039/4753 (60.08%)/(47.53%)
lr=0.1
65/100 Test set: Average loss: 28.0142/7.6700, Accuracy: 29996/4847 (59.99%)/(48.47%)
lr=0.1
66/100 Test set: Average loss: 28.0630/7.7044, Accuracy: 29978/4796 (59.96%)/(47.96%)
lr=0.1
67/100 Test set: Average loss: 27.9731/7.8488, Accuracy: 30100/4717 (60.20%)/(47.17%)
lr=0.1
68/100 Test set: Average loss: 27.8982/7.7240, Accuracy: 30053/4812 (60.11%)/(48.12%)
lr=0.1
69/100 Test set: Average loss: 27.8245/7.7493, Accuracy: 30167/4771 (60.33%)/(47.71%)
lr=0.1
```

```
70/100 Test set: Average loss: 28.0553/7.7713, Accuracy: 29886/4756 (59.77%)/(47.56%)
lr=0.1
71/100 Test set: Average loss: 27.7973/7.7865, Accuracy: 30140/4751 (60.28%)/(47.51%)
lr=0.1
72/100 Test set: Average loss: 27.7678/7.8143, Accuracy: 30190/4808 (60.38%)/(48.08%)
lr=0.1
73/100 Test set: Average loss: 27.7085/7.8173, Accuracy: 30280/4775 (60.56%)/(47.75%)
lr=0.1
74/100 Test set: Average loss: 27.5636/7.8980, Accuracy: 30358/4732 (60.72%)/(47.32%)
lr=0.1
75/100 Test set: Average loss: 27.4187/7.9372, Accuracy: 30383/4744 (60.77%)/(47.44%)
lr=0.1
76/100 Test set: Average loss: 27.6045/7.8420, Accuracy: 30404/4766 (60.81%)/(47.66%)
lr=0.1
77/100 Test set: Average loss: 27.7102/7.8855, Accuracy: 30160/4733 (60.32%)/(47.33%)
lr=0.1
78/100 Test set: Average loss: 27.3466/7.8837, Accuracy: 30451/4731 (60.90%)/(47.31%)
lr=0.1
79/100 Test set: Average loss: 27.3489/7.8783, Accuracy: 30500/4753 (61.00%)/(47.53%)
lr=0.1
80/100 Test set: Average loss: 27.5028/7.9738, Accuracy: 30428/4735 (60.86%)/(47.35%)
lr=0.1
81/100 Test set: Average loss: 27.4632/7.9054, Accuracy: 30413/4696 (60.83%)/(46.96%)
lr=0.1
82/100 Test set: Average loss: 27.2158/7.8861, Accuracy: 30587/4826 (61.17%)/(48.26%)
lr=0.1
83/100 Test set: Average loss: 27.1386/7.9933, Accuracy: 30596/4704 (61.19%)/(47.04%)
lr=0.1
84/100 Test set: Average loss: 27.5731/7.9956, Accuracy: 30320/4698 (60.64%)/(46.98%)
lr=0.1
85/100 Test set: Average loss: 27.2279/7.8320, Accuracy: 30593/4761 (61.19%)/(47.61%)
lr=0.1
86/100 Test set: Average loss: 27.1056/7.9490, Accuracy: 30589/4738 (61.18%)/(47.38%)
lr=0.1
87/100 Test set: Average loss: 27.1448/7.9649, Accuracy: 30689/4724 (61.38%)/(47.24%)
lr=0.1
88/100 Test set: Average loss: 27.1269/8.0609, Accuracy: 30597/4650 (61.19%)/(46.50%)
lr=0.1
89/100 Test set: Average loss: 27.2962/7.9731, Accuracy: 30517/4725 (61.03%)/(47.25%)
lr=0.1
90/100 Test set: Average loss: 26.9964/7.9610, Accuracy: 30806/4720 (61.61%)/(47.20%)
lr=0.1
91/100 Test set: Average loss: 26.7694/7.9557, Accuracy: 30824/4734 (61.65%)/(47.34%)
lr=0.1
92/100 Test set: Average loss: 26.8243/7.9772, Accuracy: 30863/4722 (61.73%)/(47.22%)
lr=0.1
93/100 Test set: Average loss: 26.6320/8.0725, Accuracy: 31078/4705 (62.16%)/(47.05%)
lr=0.1
94/100 Test set: Average loss: 26.7914/8.0418, Accuracy: 30874/4778 (61.75%)/(47.78%)
lr=0.1
95/100 Test set: Average loss: 26.6350/8.0672, Accuracy: 31024/4703 (62.05%)/(47.03%)
lr=0.1
96/100 Test set: Average loss: 26.7723/8.0326, Accuracy: 30804/4743 (61.61%)/(47.43%)
lr=0.1
97/100 Test set: Average loss: 26.6255/8.1150, Accuracy: 31102/4696 (62.20%)/(46.96%)
lr=0.1
98/100 Test set: Average loss: 26.7046/8.0320, Accuracy: 30965/4729 (61.93%)/(47.29%)
lr=0.1
99/100 Test set: Average loss: 26.5261/8.1510, Accuracy: 31030/4731 (62.06%)/(47.31%)
lr=0.1
```

```
Files already downloaded and verified
Files already downloaded and verified
['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'tr
uck']
(50000, 32, 32, 3)
(10000, 32, 32, 3)
DNN_MNIST_N(
  (layer1): Sequential(
    (0): Linear(in_features=3072, out_features=50, bias=True)
    (1): BatchNorm1d(50, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tr
ue)
    (2): ReLU(inplace=True)
  )
  (layer2): Sequential(
    (0): Linear(in_features=50, out_features=100, bias=True)
    (1): BatchNorm1d(100, eps=1e-05, momentum=0.1, affine=True, track_running_stats=T
rue)
    (2): ReLU(inplace=True)
  )
  (layer3): Sequential(
    (0): Linear(in_features=100, out_features=50, bias=True)
    (1): BatchNorm1d(50, eps=1e-05, momentum=0.1, affine=True, track_running_stats=Tr
ue)
    (2): ReLU(inplace=True)
  )
  (layer4): Sequential(
    (0): Linear(in_features=50, out_features=10, bias=True)
  )
)
Number of total parameters:  164710
0/100 Test set: Average loss: 47.8482/8.6701, Accuracy: 15960/3898 (31.92%)/(38.98%)
lr=0.1
1/100 Test set: Average loss: 40.3884/7.8629, Accuracy: 21149/4401 (42.30%)/(44.01%)
lr=0.1
2/100 Test set: Average loss: 37.6932/7.5085, Accuracy: 23167/4629 (46.33%)/(46.29%)
lr=0.1
3/100 Test set: Average loss: 35.7227/7.2442, Accuracy: 24495/4813 (48.99%)/(48.13%)
lr=0.1
4/100 Test set: Average loss: 34.2196/7.1697, Accuracy: 25567/4860 (51.13%)/(48.60%)
lr=0.1
5/100 Test set: Average loss: 33.1429/7.0544, Accuracy: 26367/4938 (52.73%)/(49.38%)
lr=0.1
6/100 Test set: Average loss: 32.1428/7.1014, Accuracy: 27268/4980 (54.54%)/(49.80%)
lr=0.1
7/100 Test set: Average loss: 31.3827/7.0497, Accuracy: 27713/5017 (55.43%)/(50.17%)
lr=0.1
8/100 Test set: Average loss: 30.5533/6.9971, Accuracy: 28228/5080 (56.46%)/(50.80%)
lr=0.1
9/100 Test set: Average loss: 29.7955/6.9956, Accuracy: 28785/5067 (57.57%)/(50.67%)
lr=0.1
10/100 Test set: Average loss: 29.1425/6.9749, Accuracy: 29361/5144 (58.72%)/(51.44%)
lr=0.1
11/100 Test set: Average loss: 28.6080/7.0971, Accuracy: 29670/5102 (59.34%)/(51.02%)
lr=0.1
12/100 Test set: Average loss: 28.1093/7.1131, Accuracy: 29931/5001 (59.86%)/(50.01%)
lr=0.1
13/100 Test set: Average loss: 27.6227/7.0454, Accuracy: 30379/5064 (60.76%)/(50.64%)
lr=0.1
14/100 Test set: Average loss: 27.0752/7.0140, Accuracy: 30650/5148 (61.30%)/(51.48%)
lr=0.1
```

```
15/100 Test set: Average loss: 26.5347/7.1223, Accuracy: 31150/5135 (62.30%)/(51.35%)
lr=0.1
16/100 Test set: Average loss: 26.4292/7.1353, Accuracy: 31170/5134 (62.34%)/(51.34%)
lr=0.1
17/100 Test set: Average loss: 25.9740/7.0649, Accuracy: 31531/5156 (63.06%)/(51.56%)
lr=0.1
18/100 Test set: Average loss: 25.3661/7.2336, Accuracy: 32034/5102 (64.07%)/(51.02%)
lr=0.1
19/100 Test set: Average loss: 25.1869/7.3007, Accuracy: 32062/5127 (64.12%)/(51.27%)
lr=0.1
20/100 Test set: Average loss: 24.6961/7.3345, Accuracy: 32416/5151 (64.83%)/(51.51%)
lr=0.1
21/100 Test set: Average loss: 24.2993/7.3112, Accuracy: 32702/5110 (65.40%)/(51.10%)
lr=0.1
22/100 Test set: Average loss: 23.8580/7.4011, Accuracy: 33024/5163 (66.05%)/(51.63%)
lr=0.1
23/100 Test set: Average loss: 23.4239/7.6267, Accuracy: 33247/5025 (66.49%)/(50.25%)
lr=0.1
24/100 Test set: Average loss: 23.2528/7.5800, Accuracy: 33499/5109 (67.00%)/(51.09%)
lr=0.1
25/100 Test set: Average loss: 23.2697/7.5633, Accuracy: 33448/5050 (66.90%)/(50.50%)
lr=0.1
26/100 Test set: Average loss: 22.7813/7.6354, Accuracy: 33808/5075 (67.62%)/(50.75%)
lr=0.1
27/100 Test set: Average loss: 22.4338/7.7308, Accuracy: 33937/5024 (67.87%)/(50.24%)
lr=0.1
28/100 Test set: Average loss: 22.2577/7.7846, Accuracy: 34204/5063 (68.41%)/(50.63%)
lr=0.1
29/100 Test set: Average loss: 21.8982/7.7510, Accuracy: 34406/5056 (68.81%)/(50.56%)
lr=0.1
30/100 Test set: Average loss: 21.4556/7.8769, Accuracy: 34750/5010 (69.50%)/(50.10%)
lr=0.1
31/100 Test set: Average loss: 21.1928/7.9929, Accuracy: 34918/4976 (69.84%)/(49.76%)
lr=0.1
32/100 Test set: Average loss: 21.2424/7.9418, Accuracy: 34871/5063 (69.74%)/(50.63%)
lr=0.1
33/100 Test set: Average loss: 20.8893/8.0844, Accuracy: 35051/5021 (70.10%)/(50.21%)
lr=0.1
34/100 Test set: Average loss: 20.3966/8.1125, Accuracy: 35551/5086 (71.10%)/(50.86%)
lr=0.1
35/100 Test set: Average loss: 20.4170/8.3308, Accuracy: 35557/4929 (71.11%)/(49.29%)
lr=0.1
36/100 Test set: Average loss: 20.0422/8.3053, Accuracy: 35854/5039 (71.71%)/(50.39%)
lr=0.1
37/100 Test set: Average loss: 19.7220/8.3786, Accuracy: 35998/5001 (72.00%)/(50.01%)
lr=0.1
38/100 Test set: Average loss: 19.6397/8.4947, Accuracy: 36023/4983 (72.05%)/(49.83%)
lr=0.1
39/100 Test set: Average loss: 19.5625/8.5020, Accuracy: 36130/4949 (72.26%)/(49.49%)
lr=0.1
40/100 Test set: Average loss: 18.8362/8.5643, Accuracy: 36598/4945 (73.20%)/(49.45%)
lr=0.1
41/100 Test set: Average loss: 18.7750/8.7349, Accuracy: 36652/4939 (73.30%)/(49.39%)
lr=0.1
42/100 Test set: Average loss: 18.8009/8.7103, Accuracy: 36652/4953 (73.30%)/(49.53%)
lr=0.1
43/100 Test set: Average loss: 18.4417/8.8461, Accuracy: 36983/4878 (73.97%)/(48.78%)
lr=0.1
44/100 Test set: Average loss: 18.3064/8.8135, Accuracy: 36973/4969 (73.95%)/(49.69%)
lr=0.1
```

```
45/100 Test set: Average loss: 18.1709/9.1194, Accuracy: 37022/4859 (74.04%)/(48.59%)
lr=0.1
46/100 Test set: Average loss: 18.1373/9.0221, Accuracy: 37077/4874 (74.15%)/(48.74%)
lr=0.1
47/100 Test set: Average loss: 17.8676/9.0193, Accuracy: 37293/4977 (74.59%)/(49.77%)
lr=0.1
48/100 Test set: Average loss: 17.4908/9.2521, Accuracy: 37522/4840 (75.04%)/(48.40%)
lr=0.1
49/100 Test set: Average loss: 17.5801/9.3599, Accuracy: 37571/4817 (75.14%)/(48.17%)
lr=0.1
50/100 Test set: Average loss: 17.4744/9.1882, Accuracy: 37515/4940 (75.03%)/(49.40%)
lr=0.1
51/100 Test set: Average loss: 17.1020/9.4623, Accuracy: 37784/4856 (75.57%)/(48.56%)
lr=0.1
52/100 Test set: Average loss: 17.1764/9.5635, Accuracy: 37654/4866 (75.31%)/(48.66%)
lr=0.1
53/100 Test set: Average loss: 16.7174/9.4513, Accuracy: 38010/4872 (76.02%)/(48.72%)
lr=0.1
54/100 Test set: Average loss: 16.3311/9.6214, Accuracy: 38318/4843 (76.64%)/(48.43%)
lr=0.1
55/100 Test set: Average loss: 16.2060/9.8168, Accuracy: 38541/4867 (77.08%)/(48.67%)
lr=0.1
56/100 Test set: Average loss: 16.0710/9.9152, Accuracy: 38503/4813 (77.01%)/(48.13%)
lr=0.1
57/100 Test set: Average loss: 16.5278/9.6833, Accuracy: 38258/4876 (76.52%)/(48.76%)
lr=0.1
58/100 Test set: Average loss: 16.1832/9.8687, Accuracy: 38443/4857 (76.89%)/(48.57%)
lr=0.1
59/100 Test set: Average loss: 15.8513/9.9265, Accuracy: 38751/4848 (77.50%)/(48.48%)
lr=0.1
60/100 Test set: Average loss: 15.4590/10.0427, Accuracy: 39010/4880 (78.02%)/(48.8
0%) lr=0.1
61/100 Test set: Average loss: 15.6819/10.1100, Accuracy: 38799/4817 (77.60%)/(48.1
7%) lr=0.1
62/100 Test set: Average loss: 15.3901/10.1755, Accuracy: 39045/4820 (78.09%)/(48.2
0%) lr=0.1
63/100 Test set: Average loss: 15.2798/10.3134, Accuracy: 39155/4815 (78.31%)/(48.1
5%) lr=0.1
64/100 Test set: Average loss: 15.0492/10.2601, Accuracy: 39254/4798 (78.51%)/(47.9
8%) lr=0.1
65/100 Test set: Average loss: 14.9838/10.3321, Accuracy: 39347/4816 (78.69%)/(48.1
6%) lr=0.1
66/100 Test set: Average loss: 15.1556/10.5808, Accuracy: 39157/4842 (78.31%)/(48.4
2%) lr=0.1
67/100 Test set: Average loss: 15.0404/10.5132, Accuracy: 39284/4856 (78.57%)/(48.5
6%) lr=0.1
68/100 Test set: Average loss: 14.3238/10.5193, Accuracy: 39838/4853 (79.68%)/(48.5
3%) lr=0.1
69/100 Test set: Average loss: 14.4216/10.6565, Accuracy: 39706/4751 (79.41%)/(47.5
1%) lr=0.1
70/100 Test set: Average loss: 14.3098/10.8347, Accuracy: 39768/4840 (79.54%)/(48.4
0%) lr=0.1
71/100 Test set: Average loss: 14.1162/10.8862, Accuracy: 39926/4829 (79.85%)/(48.2
9%) lr=0.1
72/100 Test set: Average loss: 13.6875/11.0363, Accuracy: 40249/4753 (80.50%)/(47.5
3%) lr=0.1
73/100 Test set: Average loss: 13.9334/11.2235, Accuracy: 39978/4751 (79.96%)/(47.5
1%) lr=0.1
74/100 Test set: Average loss: 13.9530/11.2246, Accuracy: 40055/4727 (80.11%)/(47.2
7%) lr=0.1
```

```
75/100 Test set: Average loss: 13.9326/11.2251, Accuracy: 40037/4773 (80.07%)/(47.7
3%) lr=0.1
76/100 Test set: Average loss: 14.1288/11.1396, Accuracy: 39844/4766 (79.69%)/(47.6
6%) lr=0.1
77/100 Test set: Average loss: 14.1984/11.3536, Accuracy: 39853/4808 (79.71%)/(48.0
8%) lr=0.1
78/100 Test set: Average loss: 13.2796/11.3766, Accuracy: 40525/4832 (81.05%)/(48.3
2%) lr=0.1
79/100 Test set: Average loss: 13.0723/11.3273, Accuracy: 40668/4833 (81.34%)/(48.3
3%) lr=0.1
80/100 Test set: Average loss: 12.6880/11.6593, Accuracy: 41023/4764 (82.05%)/(47.6
4%) lr=0.1
81/100 Test set: Average loss: 12.9228/11.7358, Accuracy: 40818/4753 (81.64%)/(47.5
3%) lr=0.1
82/100 Test set: Average loss: 13.0769/11.7847, Accuracy: 40575/4756 (81.15%)/(47.5
6%) lr=0.1
83/100 Test set: Average loss: 12.9574/11.7956, Accuracy: 40799/4840 (81.60%)/(48.4
0%) lr=0.1
84/100 Test set: Average loss: 12.7109/11.9412, Accuracy: 40923/4748 (81.85%)/(47.4
8%) lr=0.1
85/100 Test set: Average loss: 12.7654/11.9825, Accuracy: 40872/4749 (81.74%)/(47.4
9%) lr=0.1
86/100 Test set: Average loss: 12.3431/12.0736, Accuracy: 41185/4788 (82.37%)/(47.8
8%) lr=0.1
87/100 Test set: Average loss: 12.8464/12.0353, Accuracy: 40853/4753 (81.71%)/(47.5
3%) lr=0.1
88/100 Test set: Average loss: 12.4192/12.1610, Accuracy: 41086/4739 (82.17%)/(47.3
9%) lr=0.1
89/100 Test set: Average loss: 12.3141/12.1462, Accuracy: 41136/4775 (82.27%)/(47.7
5%) lr=0.1
90/100 Test set: Average loss: 11.8125/12.3742, Accuracy: 41654/4727 (83.31%)/(47.2
7%) lr=0.1
91/100 Test set: Average loss: 11.8646/12.4380, Accuracy: 41558/4702 (83.12%)/(47.0
2%) lr=0.1
92/100 Test set: Average loss: 12.0323/12.5159, Accuracy: 41457/4773 (82.91%)/(47.7
3%) lr=0.1
93/100 Test set: Average loss: 11.8028/12.7010, Accuracy: 41593/4723 (83.19%)/(47.2
3%) lr=0.1
94/100 Test set: Average loss: 11.7307/12.6907, Accuracy: 41713/4706 (83.43%)/(47.0
6%) lr=0.1
95/100 Test set: Average loss: 11.5351/12.6573, Accuracy: 41752/4804 (83.50%)/(48.0
4%) lr=0.1
96/100 Test set: Average loss: 11.8257/12.9166, Accuracy: 41459/4715 (82.92%)/(47.1
5%) lr=0.1
97/100 Test set: Average loss: 11.4858/12.9645, Accuracy: 41795/4709 (83.59%)/(47.0
9%) lr=0.1
98/100 Test set: Average loss: 11.5811/12.8673, Accuracy: 41784/4728 (83.57%)/(47.2
8%) lr=0.1
99/100 Test set: Average loss: 11.2398/12.9332, Accuracy: 41949/4697 (83.90%)/(46.9
7%) lr=0.1
Files already downloaded and verified
Files already downloaded and verified
['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'tr
uck']
(50000, 32, 32, 3)
(10000, 32, 32, 3)
DNN_MNIST_N(
  (layer1): Sequential(
    (0): Linear(in_features=3072, out_features=100, bias=True)
    (1): BatchNorm1d(100, eps=1e-05, momentum=0.1, affine=True, track_running_stats=T
```

```
rue)
    (2): ReLU(inplace=True)
  )
  (layer2): Sequential(
    (0): Linear(in_features=100, out_features=200, bias=True)
    (1): BatchNorm1d(200, eps=1e-05, momentum=0.1, affine=True, track_running_stats=T
rue)
    (2): ReLU(inplace=True)
  )
  (layer3): Sequential(
    (0): Linear(in_features=200, out_features=100, bias=True)
    (1): BatchNorm1d(100, eps=1e-05, momentum=0.1, affine=True, track_running_stats=T
rue)
    (2): ReLU(inplace=True)
  )
  (layer4): Sequential(
    (0): Linear(in_features=100, out_features=10, bias=True)
  )
)
Number of total parameters:  349410
0/100 Test set: Average loss: 45.4692/8.1835, Accuracy: 17593/4296 (35.19%)/(42.96%)
lr=0.1
1/100 Test set: Average loss: 38.0948/7.4219, Accuracy: 22742/4700 (45.48%)/(47.00%)
lr=0.1
2/100 Test set: Average loss: 35.1564/7.1673, Accuracy: 24783/4935 (49.57%)/(49.35%)
lr=0.1
3/100 Test set: Average loss: 33.1539/7.0839, Accuracy: 26214/4935 (52.43%)/(49.35%)
lr=0.1
4/100 Test set: Average loss: 31.6832/6.9035, Accuracy: 27423/5072 (54.85%)/(50.72%)
lr=0.1
5/100 Test set: Average loss: 30.0677/6.8897, Accuracy: 28600/5082 (57.20%)/(50.82%)
lr=0.1
6/100 Test set: Average loss: 28.8970/6.9069, Accuracy: 29368/5145 (58.74%)/(51.45%)
lr=0.1
7/100 Test set: Average loss: 27.8173/6.7820, Accuracy: 30153/5242 (60.31%)/(52.42%)
lr=0.1
8/100 Test set: Average loss: 26.8531/6.7844, Accuracy: 30868/5250 (61.74%)/(52.50%)
lr=0.1
9/100 Test set: Average loss: 25.7983/6.9828, Accuracy: 31622/5200 (63.24%)/(52.00%)
lr=0.1
10/100 Test set: Average loss: 24.8045/6.9078, Accuracy: 32217/5250 (64.43%)/(52.50%)
lr=0.1
11/100 Test set: Average loss: 23.7887/7.0802, Accuracy: 33015/5177 (66.03%)/(51.77%)
lr=0.1
12/100 Test set: Average loss: 22.7730/7.1732, Accuracy: 33751/5197 (67.50%)/(51.97%)
lr=0.1
13/100 Test set: Average loss: 22.1485/7.1998, Accuracy: 34281/5204 (68.56%)/(52.04%)
lr=0.1
14/100 Test set: Average loss: 21.2610/7.4267, Accuracy: 34863/5160 (69.73%)/(51.60%)
lr=0.1
15/100 Test set: Average loss: 20.3746/7.5969, Accuracy: 35502/5165 (71.00%)/(51.65%)
lr=0.1
16/100 Test set: Average loss: 19.7653/7.8315, Accuracy: 36023/5098 (72.05%)/(50.98%)
lr=0.1
17/100 Test set: Average loss: 19.0778/7.9572, Accuracy: 36408/5090 (72.82%)/(50.90%)
lr=0.1
18/100 Test set: Average loss: 18.5201/8.0979, Accuracy: 36725/5039 (73.45%)/(50.39%)
lr=0.1
19/100 Test set: Average loss: 17.5618/8.0325, Accuracy: 37618/5183 (75.24%)/(51.83%)
lr=0.1
```

```
20/100 Test set: Average loss: 16.9794/8.2724, Accuracy: 37987/5114 (75.97%)/(51.14%)
lr=0.1
21/100 Test set: Average loss: 16.5356/8.4628, Accuracy: 38130/5097 (76.26%)/(50.97%)
lr=0.1
22/100 Test set: Average loss: 15.7442/8.4382, Accuracy: 38865/5077 (77.73%)/(50.77%)
lr=0.1
23/100 Test set: Average loss: 15.2333/8.7188, Accuracy: 39260/5090 (78.52%)/(50.90%)
lr=0.1
24/100 Test set: Average loss: 14.5850/8.9875, Accuracy: 39620/5006 (79.24%)/(50.06%)
lr=0.1
25/100 Test set: Average loss: 14.0725/9.2255, Accuracy: 40034/5082 (80.07%)/(50.82%)
lr=0.1
26/100 Test set: Average loss: 13.4591/9.2082, Accuracy: 40517/5109 (81.03%)/(51.09%)
lr=0.1
27/100 Test set: Average loss: 12.8215/9.5065, Accuracy: 40898/5081 (81.80%)/(50.81%)
lr=0.1
28/100 Test set: Average loss: 12.4307/9.5313, Accuracy: 41237/5014 (82.47%)/(50.14%)
lr=0.1
29/100 Test set: Average loss: 11.7865/10.0029, Accuracy: 41743/4962 (83.49%)/(49.6
2%) lr=0.1
30/100 Test set: Average loss: 11.2157/10.1751, Accuracy: 42034/5092 (84.07%)/(50.9
2%) lr=0.1
31/100 Test set: Average loss: 10.7100/10.1976, Accuracy: 42458/5070 (84.92%)/(50.7
0%) lr=0.1
32/100 Test set: Average loss: 10.1687/10.4945, Accuracy: 42854/5059 (85.71%)/(50.5
9%) lr=0.1
33/100 Test set: Average loss: 10.1461/10.5820, Accuracy: 42834/5055 (85.67%)/(50.5
5%) lr=0.1
34/100 Test set: Average loss: 10.0289/10.6831, Accuracy: 42944/5061 (85.89%)/(50.6
1%) lr=0.1
35/100 Test set: Average loss: 9.6061/11.1965, Accuracy: 43264/4998 (86.53%)/(49.98%)
lr=0.1
36/100 Test set: Average loss: 8.9153/11.3485, Accuracy: 43716/4985 (87.43%)/(49.85%)
lr=0.1
37/100 Test set: Average loss: 8.6709/11.2435, Accuracy: 44004/5077 (88.01%)/(50.77%)
lr=0.1
38/100 Test set: Average loss: 8.1678/11.7211, Accuracy: 44327/4944 (88.65%)/(49.44%)
lr=0.1
39/100 Test set: Average loss: 7.7046/12.0453, Accuracy: 44704/5001 (89.41%)/(50.01%)
lr=0.1
40/100 Test set: Average loss: 7.5524/12.3266, Accuracy: 44716/4995 (89.43%)/(49.95%)
lr=0.1
41/100 Test set: Average loss: 7.2343/12.3361, Accuracy: 45016/5009 (90.03%)/(50.09%)
lr=0.1
42/100 Test set: Average loss: 7.2674/12.8278, Accuracy: 44913/5007 (89.83%)/(50.07%)
lr=0.1
43/100 Test set: Average loss: 7.2606/12.8579, Accuracy: 44901/4979 (89.80%)/(49.79%)
lr=0.1
44/100 Test set: Average loss: 6.5720/12.8033, Accuracy: 45505/5008 (91.01%)/(50.08%)
lr=0.1
45/100 Test set: Average loss: 5.8498/13.0753, Accuracy: 45971/4956 (91.94%)/(49.56%)
lr=0.1
46/100 Test set: Average loss: 5.7083/13.4398, Accuracy: 46140/5019 (92.28%)/(50.19%)
lr=0.1
47/100 Test set: Average loss: 5.2068/13.6026, Accuracy: 46425/4987 (92.85%)/(49.87%)
lr=0.1
48/100 Test set: Average loss: 4.9125/13.9184, Accuracy: 46699/4968 (93.40%)/(49.68%)
lr=0.1
49/100 Test set: Average loss: 5.1483/14.0646, Accuracy: 46482/4980 (92.96%)/(49.80%)
lr=0.1
```

```
50/100 Test set: Average loss: 5.0299/14.2688, Accuracy: 46627/5042 (93.25%)/(50.42%)
lr=0.1
51/100 Test set: Average loss: 4.8544/14.3432, Accuracy: 46777/4948 (93.55%)/(49.48%)
lr=0.1
52/100 Test set: Average loss: 4.5202/14.8043, Accuracy: 47002/4908 (94.00%)/(49.08%)
lr=0.1
53/100 Test set: Average loss: 4.3216/15.0370, Accuracy: 47126/4980 (94.25%)/(49.80%)
lr=0.1
54/100 Test set: Average loss: 3.9575/15.1437, Accuracy: 47434/4936 (94.87%)/(49.36%)
lr=0.1
55/100 Test set: Average loss: 4.1727/15.2985, Accuracy: 47181/4964 (94.36%)/(49.64%)
lr=0.1
56/100 Test set: Average loss: 4.4242/15.4926, Accuracy: 46891/4939 (93.78%)/(49.39%)
lr=0.1
57/100 Test set: Average loss: 4.3541/15.4901, Accuracy: 47024/5016 (94.05%)/(50.16%)
lr=0.1
58/100 Test set: Average loss: 3.9504/15.6504, Accuracy: 47308/4900 (94.62%)/(49.00%)
lr=0.1
59/100 Test set: Average loss: 3.4234/15.7844, Accuracy: 47759/4984 (95.52%)/(49.84%)
lr=0.1
60/100 Test set: Average loss: 3.0284/16.0085, Accuracy: 48049/5062 (96.10%)/(50.62%)
lr=0.1
61/100 Test set: Average loss: 2.7909/16.3850, Accuracy: 48273/4954 (96.55%)/(49.54%)
lr=0.1
62/100 Test set: Average loss: 2.8690/16.6961, Accuracy: 48175/4937 (96.35%)/(49.37%)
lr=0.1
63/100 Test set: Average loss: 2.6229/16.8052, Accuracy: 48340/4937 (96.68%)/(49.37%)
lr=0.1
64/100 Test set: Average loss: 2.2254/16.9077, Accuracy: 48682/4971 (97.36%)/(49.71%)
lr=0.1
65/100 Test set: Average loss: 1.8381/17.1234, Accuracy: 49020/4991 (98.04%)/(49.91%)
lr=0.1
66/100 Test set: Average loss: 1.6739/17.6264, Accuracy: 49098/5065 (98.20%)/(50.65%)
lr=0.1
67/100 Test set: Average loss: 1.5274/17.6100, Accuracy: 49214/4978 (98.43%)/(49.78%)
lr=0.1
68/100 Test set: Average loss: 1.4802/18.1671, Accuracy: 49231/4950 (98.46%)/(49.50%)
lr=0.1
69/100 Test set: Average loss: 1.4214/18.2231, Accuracy: 49275/5004 (98.55%)/(50.04%)
lr=0.1
70/100 Test set: Average loss: 1.2084/18.2932, Accuracy: 49422/5026 (98.84%)/(50.26%)
lr=0.1
71/100 Test set: Average loss: 1.1164/18.6322, Accuracy: 49507/4994 (99.01%)/(49.94%)
lr=0.1
72/100 Test set: Average loss: 1.0102/18.9400, Accuracy: 49529/5022 (99.06%)/(50.22%)
lr=0.1
73/100 Test set: Average loss: 0.9752/18.9338, Accuracy: 49588/5050 (99.18%)/(50.50%)
lr=0.1
74/100 Test set: Average loss: 0.8526/19.1882, Accuracy: 49646/4950 (99.29%)/(49.50%)
lr=0.1
75/100 Test set: Average loss: 0.8255/19.4710, Accuracy: 49679/4935 (99.36%)/(49.35%)
lr=0.1
76/100 Test set: Average loss: 0.9511/19.6692, Accuracy: 49540/4960 (99.08%)/(49.60%)
lr=0.1
77/100 Test set: Average loss: 0.8458/20.0287, Accuracy: 49627/4906 (99.25%)/(49.06%)
lr=0.1
78/100 Test set: Average loss: 0.8211/20.0577, Accuracy: 49657/4967 (99.31%)/(49.67%)
lr=0.1
79/100 Test set: Average loss: 0.7668/20.2295, Accuracy: 49693/4988 (99.39%)/(49.88%)
lr=0.1
```

```
80/100 Test set: Average loss: 0.8363/20.4920, Accuracy: 49604/4962 (99.21%)/(49.62%)
lr=0.1
81/100 Test set: Average loss: 0.9081/20.6149, Accuracy: 49561/4975 (99.12%)/(49.75%)
lr=0.1
82/100 Test set: Average loss: 0.8654/20.5450, Accuracy: 49581/4957 (99.16%)/(49.57%)
lr=0.1
83/100 Test set: Average loss: 0.6855/20.8152, Accuracy: 49708/4921 (99.42%)/(49.21%)
lr=0.1
84/100 Test set: Average loss: 0.6386/20.9013, Accuracy: 49750/5004 (99.50%)/(50.04%)
lr=0.1
85/100 Test set: Average loss: 0.6104/21.0078, Accuracy: 49750/4969 (99.50%)/(49.69%)
lr=0.1
86/100 Test set: Average loss: 0.5772/21.2201, Accuracy: 49775/4994 (99.55%)/(49.94%)
lr=0.1
87/100 Test set: Average loss: 0.5395/21.2634, Accuracy: 49799/4989 (99.60%)/(49.89%)
lr=0.1
88/100 Test set: Average loss: 0.4376/21.3380, Accuracy: 49862/4970 (99.72%)/(49.70%)
lr=0.1
89/100 Test set: Average loss: 0.3603/21.4416, Accuracy: 49912/4964 (99.82%)/(49.64%)
lr=0.1
90/100 Test set: Average loss: 0.2638/21.6500, Accuracy: 49955/4996 (99.91%)/(49.96%)
lr=0.1
91/100 Test set: Average loss: 0.2080/21.6689, Accuracy: 49978/5003 (99.96%)/(50.03%)
lr=0.1
92/100 Test set: Average loss: 0.1531/21.8894, Accuracy: 49988/4972 (99.98%)/(49.72%)
lr=0.1
93/100 Test set: Average loss: 0.1131/22.0066, Accuracy: 49995/5033 (99.99%)/(50.33%)
lr=0.1
94/100 Test set: Average loss: 0.0867/22.1729, Accuracy: 49999/5015 (100.00%)/(50.1
5%) lr=0.1
95/100 Test set: Average loss: 0.0732/22.2273, Accuracy: 49999/5022 (100.00%)/(50.2
2%) lr=0.1
96/100 Test set: Average loss: 0.0655/22.3252, Accuracy: 50000/5027 (100.00%)/(50.2
7%) lr=0.1
97/100 Test set: Average loss: 0.0594/22.3775, Accuracy: 50000/5015 (100.00%)/(50.1
5%) lr=0.1
98/100 Test set: Average loss: 0.0564/22.5187, Accuracy: 50000/5012 (100.00%)/(50.1
2%) lr=0.1
99/100 Test set: Average loss: 0.0527/22.5677, Accuracy: 50000/5021 (100.00%)/(50.2
1%) lr=0.1
Files already downloaded and verified
Files already downloaded and verified
['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'tr
uck']
(50000, 32, 32, 3)
(10000, 32, 32, 3)
DNN_MNIST_N(
  (layer1): Sequential(
    (0): Linear(in_features=3072, out_features=200, bias=True)
    (1): BatchNorm1d(200, eps=1e-05, momentum=0.1, affine=True, track_running_stats=T
rue)
    (2): ReLU(inplace=True)
  )
  (layer2): Sequential(
    (0): Linear(in_features=200, out_features=400, bias=True)
    (1): BatchNorm1d(400, eps=1e-05, momentum=0.1, affine=True, track_running_stats=T
rue)
    (2): ReLU(inplace=True)
  )
  (layer3): Sequential(
```

```
    (0): Linear(in_features=400, out_features=200, bias=True)
    (1): BatchNorm1d(200, eps=1e-05, momentum=0.1, affine=True, track_running_stats=T
rue)
    (2): ReLU(inplace=True)
  )
  (layer4): Sequential(
    (0): Linear(in_features=200, out_features=10, bias=True)
  )
)
Number of total parameters:  778810
0/100 Test set: Average loss: 43.5270/7.8528, Accuracy: 18849/4440 (37.70%)/(44.40%)
lr=0.1
1/100 Test set: Average loss: 36.3769/7.2862, Accuracy: 24033/4875 (48.07%)/(48.75%)
lr=0.1
2/100 Test set: Average loss: 33.1127/6.9791, Accuracy: 26483/5035 (52.97%)/(50.35%)
lr=0.1
3/100 Test set: Average loss: 30.8535/6.7174, Accuracy: 27922/5200 (55.84%)/(52.00%)
lr=0.1
4/100 Test set: Average loss: 28.8840/6.6715, Accuracy: 29470/5302 (58.94%)/(53.02%)
lr=0.1
5/100 Test set: Average loss: 27.3343/6.7296, Accuracy: 30609/5301 (61.22%)/(53.01%)
lr=0.1
6/100 Test set: Average loss: 25.3655/6.8122, Accuracy: 31839/5232 (63.68%)/(52.32%)
lr=0.1
7/100 Test set: Average loss: 23.7855/6.9992, Accuracy: 33114/5342 (66.23%)/(53.42%)
lr=0.1
8/100 Test set: Average loss: 22.4843/6.9071, Accuracy: 34048/5277 (68.10%)/(52.77%)
lr=0.1
9/100 Test set: Average loss: 20.8687/7.4274, Accuracy: 35235/5300 (70.47%)/(53.00%)
lr=0.1
10/100 Test set: Average loss: 19.9303/7.3745, Accuracy: 35894/5247 (71.79%)/(52.47%)
lr=0.1
11/100 Test set: Average loss: 18.2880/7.7651, Accuracy: 37121/5208 (74.24%)/(52.08%)
lr=0.1
12/100 Test set: Average loss: 17.0712/7.5185, Accuracy: 38024/5334 (76.05%)/(53.34%)
lr=0.1
13/100 Test set: Average loss: 15.9866/8.1705, Accuracy: 38678/5170 (77.36%)/(51.70%)
lr=0.1
14/100 Test set: Average loss: 15.2703/8.4529, Accuracy: 39282/5113 (78.56%)/(51.13%)
lr=0.1
15/100 Test set: Average loss: 13.5173/8.2730, Accuracy: 40608/5344 (81.22%)/(53.44%)
lr=0.1
16/100 Test set: Average loss: 12.2237/8.4828, Accuracy: 41437/5269 (82.87%)/(52.69%)
lr=0.1
17/100 Test set: Average loss: 11.0328/8.9851, Accuracy: 42380/5291 (84.76%)/(52.91%)
lr=0.1
18/100 Test set: Average loss: 10.6289/9.0309, Accuracy: 42659/5228 (85.32%)/(52.28%)
lr=0.1
19/100 Test set: Average loss: 9.4822/9.6929, Accuracy: 43451/5218 (86.90%)/(52.18%)
lr=0.1
20/100 Test set: Average loss: 8.1458/9.8204, Accuracy: 44572/5193 (89.14%)/(51.93%)
lr=0.1
21/100 Test set: Average loss: 7.5104/10.4417, Accuracy: 44863/5261 (89.73%)/(52.61%)
lr=0.1
22/100 Test set: Average loss: 7.2563/10.3858, Accuracy: 45033/5236 (90.07%)/(52.36%)
lr=0.1
23/100 Test set: Average loss: 6.1280/10.5669, Accuracy: 45894/5270 (91.79%)/(52.70%)
lr=0.1
24/100 Test set: Average loss: 5.7958/11.0829, Accuracy: 46102/5240 (92.20%)/(52.40%)
lr=0.1
```

```
25/100 Test set: Average loss: 5.4052/11.4065, Accuracy: 46288/5289 (92.58%)/(52.89%)
lr=0.1
26/100 Test set: Average loss: 4.6702/11.3662, Accuracy: 46924/5267 (93.85%)/(52.67%)
lr=0.1
27/100 Test set: Average loss: 3.9450/11.8341, Accuracy: 47483/5263 (94.97%)/(52.63%)
lr=0.1
28/100 Test set: Average loss: 3.6940/11.8693, Accuracy: 47654/5325 (95.31%)/(53.25%)
lr=0.1
29/100 Test set: Average loss: 3.3740/12.2038, Accuracy: 47853/5301 (95.71%)/(53.01%)
lr=0.1
30/100 Test set: Average loss: 2.6763/12.6078, Accuracy: 48392/5227 (96.78%)/(52.27%)
lr=0.1
31/100 Test set: Average loss: 2.3161/13.4037, Accuracy: 48677/5249 (97.35%)/(52.49%)
lr=0.1
32/100 Test set: Average loss: 2.0052/13.1507, Accuracy: 48868/5306 (97.74%)/(53.06%)
lr=0.1
33/100 Test set: Average loss: 1.7160/13.6416, Accuracy: 49096/5250 (98.19%)/(52.50%)
lr=0.1
34/100 Test set: Average loss: 1.5268/13.7040, Accuracy: 49188/5302 (98.38%)/(53.02%)
lr=0.1
35/100 Test set: Average loss: 1.2557/13.9164, Accuracy: 49361/5359 (98.72%)/(53.59%)
lr=0.1
36/100 Test set: Average loss: 0.9639/13.9751, Accuracy: 49620/5331 (99.24%)/(53.31%)
lr=0.1
37/100 Test set: Average loss: 0.6824/14.4444, Accuracy: 49773/5294 (99.55%)/(52.94%)
lr=0.1
38/100 Test set: Average loss: 0.4865/14.6000, Accuracy: 49862/5385 (99.72%)/(53.85%)
lr=0.1
39/100 Test set: Average loss: 0.3897/14.7737, Accuracy: 49892/5386 (99.78%)/(53.86%)
lr=0.1
40/100 Test set: Average loss: 0.3299/14.9097, Accuracy: 49922/5346 (99.84%)/(53.46%)
lr=0.1
41/100 Test set: Average loss: 0.2358/15.1058, Accuracy: 49962/5371 (99.92%)/(53.71%)
lr=0.1
42/100 Test set: Average loss: 0.1743/15.2933, Accuracy: 49973/5344 (99.95%)/(53.44%)
lr=0.1
43/100 Test set: Average loss: 0.1345/15.2943, Accuracy: 49986/5408 (99.97%)/(54.08%)
lr=0.1
44/100 Test set: Average loss: 0.0811/15.4393, Accuracy: 50000/5383 (100.00%)/(53.8
3%) lr=0.1
45/100 Test set: Average loss: 0.0631/15.5597, Accuracy: 50000/5367 (100.00%)/(53.6
7%) lr=0.1
46/100 Test set: Average loss: 0.0537/15.7060, Accuracy: 50000/5391 (100.00%)/(53.9
1%) lr=0.1
47/100 Test set: Average loss: 0.0494/15.7730, Accuracy: 50000/5393 (100.00%)/(53.9
3%) lr=0.1
48/100 Test set: Average loss: 0.0451/15.8972, Accuracy: 50000/5364 (100.00%)/(53.6
4%) lr=0.1
49/100 Test set: Average loss: 0.0424/16.0023, Accuracy: 50000/5379 (100.00%)/(53.7
9%) lr=0.1
50/100 Test set: Average loss: 0.0399/16.0862, Accuracy: 50000/5395 (100.00%)/(53.9
5%) lr=0.1
51/100 Test set: Average loss: 0.0388/16.1758, Accuracy: 50000/5390 (100.00%)/(53.9
0%) lr=0.1
52/100 Test set: Average loss: 0.0370/16.2590, Accuracy: 50000/5387 (100.00%)/(53.8
7%) lr=0.1
53/100 Test set: Average loss: 0.0346/16.3709, Accuracy: 50000/5390 (100.00%)/(53.9
0%) lr=0.1
54/100 Test set: Average loss: 0.0323/16.4152, Accuracy: 50000/5381 (100.00%)/(53.8
1%) lr=0.1
```

```
55/100 Test set: Average loss: 0.0313/16.5365, Accuracy: 50000/5380 (100.00%)/(53.8
0%) lr=0.1
56/100 Test set: Average loss: 0.0301/16.5719, Accuracy: 50000/5373 (100.00%)/(53.7
3%) lr=0.1
57/100 Test set: Average loss: 0.0289/16.6324, Accuracy: 50000/5386 (100.00%)/(53.8
6%) lr=0.1
58/100 Test set: Average loss: 0.0288/16.7170, Accuracy: 50000/5367 (100.00%)/(53.6
7%) lr=0.1
59/100 Test set: Average loss: 0.0271/16.7393, Accuracy: 50000/5388 (100.00%)/(53.8
8%) lr=0.1
60/100 Test set: Average loss: 0.0261/16.8012, Accuracy: 50000/5370 (100.00%)/(53.7
0%) lr=0.1
61/100 Test set: Average loss: 0.0252/16.8862, Accuracy: 50000/5358 (100.00%)/(53.5
8%) lr=0.1
62/100 Test set: Average loss: 0.0246/16.9281, Accuracy: 50000/5346 (100.00%)/(53.4
6%) lr=0.1
63/100 Test set: Average loss: 0.0241/16.9932, Accuracy: 50000/5361 (100.00%)/(53.6
1%) lr=0.1
64/100 Test set: Average loss: 0.0228/17.0566, Accuracy: 50000/5368 (100.00%)/(53.6
8%) lr=0.1
65/100 Test set: Average loss: 0.0227/17.1191, Accuracy: 50000/5366 (100.00%)/(53.6
6%) lr=0.1
66/100 Test set: Average loss: 0.0220/17.1646, Accuracy: 50000/5366 (100.00%)/(53.6
6%) lr=0.1
67/100 Test set: Average loss: 0.0210/17.2242, Accuracy: 50000/5372 (100.00%)/(53.7
2%) lr=0.1
68/100 Test set: Average loss: 0.0203/17.2710, Accuracy: 50000/5359 (100.00%)/(53.5
9%) lr=0.1
69/100 Test set: Average loss: 0.0197/17.3078, Accuracy: 50000/5367 (100.00%)/(53.6
7%) lr=0.1
70/100 Test set: Average loss: 0.0193/17.3642, Accuracy: 50000/5359 (100.00%)/(53.5
9%) lr=0.1
71/100 Test set: Average loss: 0.0195/17.4074, Accuracy: 50000/5356 (100.00%)/(53.5
6%) lr=0.1
72/100 Test set: Average loss: 0.0187/17.4430, Accuracy: 50000/5368 (100.00%)/(53.6
8%) lr=0.1
73/100 Test set: Average loss: 0.0186/17.4780, Accuracy: 50000/5360 (100.00%)/(53.6
0%) lr=0.1
74/100 Test set: Average loss: 0.0182/17.5527, Accuracy: 50000/5362 (100.00%)/(53.6
2%) lr=0.1
75/100 Test set: Average loss: 0.0173/17.5619, Accuracy: 50000/5352 (100.00%)/(53.5
2%) lr=0.1
76/100 Test set: Average loss: 0.0175/17.5926, Accuracy: 50000/5360 (100.00%)/(53.6
0%) lr=0.1
77/100 Test set: Average loss: 0.0170/17.6789, Accuracy: 50000/5354 (100.00%)/(53.5
4%) lr=0.1
78/100 Test set: Average loss: 0.0167/17.7135, Accuracy: 50000/5349 (100.00%)/(53.4
9%) lr=0.1
79/100 Test set: Average loss: 0.0160/17.7511, Accuracy: 50000/5359 (100.00%)/(53.5
9%) lr=0.1
80/100 Test set: Average loss: 0.0160/17.7700, Accuracy: 50000/5345 (100.00%)/(53.4
5%) lr=0.1
81/100 Test set: Average loss: 0.0153/17.8185, Accuracy: 50000/5363 (100.00%)/(53.6
3%) lr=0.1
82/100 Test set: Average loss: 0.0153/17.8627, Accuracy: 50000/5364 (100.00%)/(53.6
4%) lr=0.1
83/100 Test set: Average loss: 0.0152/17.8986, Accuracy: 50000/5353 (100.00%)/(53.5
3%) lr=0.1
84/100 Test set: Average loss: 0.0145/17.9423, Accuracy: 50000/5358 (100.00%)/(53.5
8%) lr=0.1
```

```
85/100 Test set: Average loss: 0.0142/17.9535, Accuracy: 50000/5348 (100.00%)/(53.4
8%) lr=0.1
86/100 Test set: Average loss: 0.0142/18.0002, Accuracy: 50000/5341 (100.00%)/(53.4
1%) lr=0.1
87/100 Test set: Average loss: 0.0137/18.0455, Accuracy: 50000/5352 (100.00%)/(53.5
2%) lr=0.1
88/100 Test set: Average loss: 0.0137/18.0890, Accuracy: 50000/5345 (100.00%)/(53.4
5%) lr=0.1
89/100 Test set: Average loss: 0.0133/18.1137, Accuracy: 50000/5355 (100.00%)/(53.5
5%) lr=0.1
90/100 Test set: Average loss: 0.0127/18.1153, Accuracy: 50000/5348 (100.00%)/(53.4
8%) lr=0.1
91/100 Test set: Average loss: 0.0127/18.1553, Accuracy: 50000/5348 (100.00%)/(53.4
8%) lr=0.1
92/100 Test set: Average loss: 0.0125/18.1864, Accuracy: 50000/5349 (100.00%)/(53.4
9%) lr=0.1
93/100 Test set: Average loss: 0.0126/18.2250, Accuracy: 50000/5351 (100.00%)/(53.5
1%) lr=0.1
94/100 Test set: Average loss: 0.0125/18.2675, Accuracy: 50000/5352 (100.00%)/(53.5
2%) lr=0.1
95/100 Test set: Average loss: 0.0120/18.2952, Accuracy: 50000/5350 (100.00%)/(53.5
0%) lr=0.1
96/100 Test set: Average loss: 0.0122/18.3172, Accuracy: 50000/5335 (100.00%)/(53.3
5%) lr=0.1
97/100 Test set: Average loss: 0.0118/18.3561, Accuracy: 50000/5350 (100.00%)/(53.5
0%) lr=0.1
98/100 Test set: Average loss: 0.0117/18.3927, Accuracy: 50000/5343 (100.00%)/(53.4
3%) lr=0.1
99/100 Test set: Average loss: 0.0113/18.4125, Accuracy: 50000/5345 (100.00%)/(53.4
5%) lr=0.1
Files already downloaded and verified
Files already downloaded and verified
['airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'tr
uck']
(50000, 32, 32, 3)
(10000, 32, 32, 3)
DNN_MNIST_N(
  (layer1): Sequential(
    (0): Linear(in_features=3072, out_features=500, bias=True)
    (1): BatchNorm1d(500, eps=1e-05, momentum=0.1, affine=True, track_running_stats=T
rue)
    (2): ReLU(inplace=True)
  )
  (layer2): Sequential(
    (0): Linear(in_features=500, out_features=1000, bias=True)
    (1): BatchNorm1d(1000, eps=1e-05, momentum=0.1, affine=True, track_running_stats=
True)
    (2): ReLU(inplace=True)
  )
  (layer3): Sequential(
    (0): Linear(in_features=1000, out_features=500, bias=True)
    (1): BatchNorm1d(500, eps=1e-05, momentum=0.1, affine=True, track_running_stats=T
rue)
    (2): ReLU(inplace=True)
  )
  (layer4): Sequential(
    (0): Linear(in_features=500, out_features=10, bias=True)
  )
)
Number of total parameters:  2547010
```

```
0/100 Test set: Average loss: 42.5347/7.8030, Accuracy: 19650/4501 (39.30%)/(45.01%)
lr=0.1
1/100 Test set: Average loss: 34.7181/7.0182, Accuracy: 25180/4971 (50.36%)/(49.71%)
lr=0.1
2/100 Test set: Average loss: 31.2642/6.8359, Accuracy: 27544/5168 (55.09%)/(51.68%)
lr=0.1
3/100 Test set: Average loss: 28.5049/6.8569, Accuracy: 29619/5174 (59.24%)/(51.74%)
lr=0.1
4/100 Test set: Average loss: 26.1240/6.8331, Accuracy: 31544/5298 (63.09%)/(52.98%)
lr=0.1
5/100 Test set: Average loss: 23.6315/6.9949, Accuracy: 33254/5313 (66.51%)/(53.13%)
lr=0.1
6/100 Test set: Average loss: 21.5378/7.1202, Accuracy: 34778/5296 (69.56%)/(52.96%)
lr=0.1
7/100 Test set: Average loss: 19.8794/7.5931, Accuracy: 35955/5124 (71.91%)/(51.24%)
lr=0.1
8/100 Test set: Average loss: 17.8926/7.2238, Accuracy: 37433/5472 (74.87%)/(54.72%)
lr=0.1
9/100 Test set: Average loss: 15.2043/7.6370, Accuracy: 39465/5383 (78.93%)/(53.83%)
lr=0.1
10/100 Test set: Average loss: 13.5703/8.0411, Accuracy: 40492/5292 (80.98%)/(52.92%)
lr=0.1
11/100 Test set: Average loss: 11.7862/8.3575, Accuracy: 41871/5401 (83.74%)/(54.01%)
lr=0.1
12/100 Test set: Average loss: 10.4070/8.8104, Accuracy: 42742/5320 (85.48%)/(53.20%)
lr=0.1
13/100 Test set: Average loss: 8.0847/9.3076, Accuracy: 44533/5305 (89.07%)/(53.05%)
lr=0.1
14/100 Test set: Average loss: 7.3221/9.4731, Accuracy: 45007/5451 (90.01%)/(54.51%)
lr=0.1
15/100 Test set: Average loss: 6.0451/9.6966, Accuracy: 46002/5378 (92.00%)/(53.78%)
lr=0.1
16/100 Test set: Average loss: 5.3755/9.8033, Accuracy: 46428/5483 (92.86%)/(54.83%)
lr=0.1
17/100 Test set: Average loss: 4.3830/10.2782, Accuracy: 47171/5337 (94.34%)/(53.37%)
lr=0.1
18/100 Test set: Average loss: 3.5398/10.4180, Accuracy: 47817/5469 (95.63%)/(54.69%)
lr=0.1
19/100 Test set: Average loss: 2.5708/10.8724, Accuracy: 48501/5458 (97.00%)/(54.58%)
lr=0.1
20/100 Test set: Average loss: 2.0040/10.8302, Accuracy: 48852/5550 (97.70%)/(55.50%)
lr=0.1
21/100 Test set: Average loss: 1.5800/11.3681, Accuracy: 49161/5490 (98.32%)/(54.90%)
lr=0.1
22/100 Test set: Average loss: 1.0931/11.4620, Accuracy: 49485/5570 (98.97%)/(55.70%)
lr=0.1
23/100 Test set: Average loss: 0.7773/11.7482, Accuracy: 49685/5590 (99.37%)/(55.90%)
lr=0.1
24/100 Test set: Average loss: 0.5360/12.0281, Accuracy: 49812/5609 (99.62%)/(56.09%)
lr=0.1
25/100 Test set: Average loss: 0.3388/12.0893, Accuracy: 49908/5649 (99.82%)/(56.49%)
lr=0.1
26/100 Test set: Average loss: 0.2238/12.2477, Accuracy: 49940/5652 (99.88%)/(56.52%)
lr=0.1
27/100 Test set: Average loss: 0.1464/12.2717, Accuracy: 49975/5678 (99.95%)/(56.78%)
lr=0.1
28/100 Test set: Average loss: 0.1033/12.4038, Accuracy: 49989/5712 (99.98%)/(57.12%)
lr=0.1
29/100 Test set: Average loss: 0.0595/12.4652, Accuracy: 50000/5739 (100.00%)/(57.3
9%) lr=0.1
```

```
30/100 Test set: Average loss: 0.0462/12.5998, Accuracy: 49997/5756 (99.99%)/(57.56%)
lr=0.1
31/100 Test set: Average loss: 0.0352/12.6484, Accuracy: 50000/5764 (100.00%)/(57.6
4%) lr=0.1
32/100 Test set: Average loss: 0.0295/12.7098, Accuracy: 50000/5767 (100.00%)/(57.6
7%) lr=0.1
33/100 Test set: Average loss: 0.0263/12.8021, Accuracy: 50000/5773 (100.00%)/(57.7
3%) lr=0.1
34/100 Test set: Average loss: 0.0242/12.8773, Accuracy: 50000/5762 (100.00%)/(57.6
2%) lr=0.1
35/100 Test set: Average loss: 0.0226/12.9522, Accuracy: 50000/5752 (100.00%)/(57.5
2%) lr=0.1
36/100 Test set: Average loss: 0.0211/13.0171, Accuracy: 50000/5758 (100.00%)/(57.5
8%) lr=0.1
37/100 Test set: Average loss: 0.0200/13.0623, Accuracy: 50000/5738 (100.00%)/(57.3
8%) lr=0.1
38/100 Test set: Average loss: 0.0189/13.1314, Accuracy: 50000/5752 (100.00%)/(57.5
2%) lr=0.1
39/100 Test set: Average loss: 0.0181/13.1920, Accuracy: 50000/5749 (100.00%)/(57.4
9%) lr=0.1
40/100 Test set: Average loss: 0.0173/13.2413, Accuracy: 50000/5748 (100.00%)/(57.4
8%) lr=0.1
41/100 Test set: Average loss: 0.0164/13.3128, Accuracy: 50000/5751 (100.00%)/(57.5
1%) lr=0.1
42/100 Test set: Average loss: 0.0158/13.3541, Accuracy: 50000/5737 (100.00%)/(57.3
7%) lr=0.1
43/100 Test set: Average loss: 0.0151/13.3943, Accuracy: 50000/5751 (100.00%)/(57.5
1%) lr=0.1
44/100 Test set: Average loss: 0.0147/13.4309, Accuracy: 50000/5734 (100.00%)/(57.3
4%) lr=0.1
45/100 Test set: Average loss: 0.0142/13.4843, Accuracy: 50000/5752 (100.00%)/(57.5
2%) lr=0.1
46/100 Test set: Average loss: 0.0136/13.5216, Accuracy: 50000/5747 (100.00%)/(57.4
7%) lr=0.1
47/100 Test set: Average loss: 0.0131/13.5611, Accuracy: 50000/5739 (100.00%)/(57.3
9%) lr=0.1
48/100 Test set: Average loss: 0.0127/13.5959, Accuracy: 50000/5740 (100.00%)/(57.4
0%) lr=0.1
49/100 Test set: Average loss: 0.0122/13.6349, Accuracy: 50000/5736 (100.00%)/(57.3
6%) lr=0.1
50/100 Test set: Average loss: 0.0122/13.6863, Accuracy: 50000/5744 (100.00%)/(57.4
4%) lr=0.1
51/100 Test set: Average loss: 0.0117/13.7120, Accuracy: 50000/5735 (100.00%)/(57.3
5%) lr=0.1
52/100 Test set: Average loss: 0.0112/13.7487, Accuracy: 50000/5741 (100.00%)/(57.4
1%) lr=0.1
53/100 Test set: Average loss: 0.0110/13.7793, Accuracy: 50000/5740 (100.00%)/(57.4
0%) lr=0.1
54/100 Test set: Average loss: 0.0107/13.8149, Accuracy: 50000/5738 (100.00%)/(57.3
8%) lr=0.1
55/100 Test set: Average loss: 0.0102/13.8530, Accuracy: 50000/5739 (100.00%)/(57.3
9%) lr=0.1
56/100 Test set: Average loss: 0.0102/13.8758, Accuracy: 50000/5732 (100.00%)/(57.3
2%) lr=0.1
57/100 Test set: Average loss: 0.0099/13.9006, Accuracy: 50000/5730 (100.00%)/(57.3
0%) lr=0.1
58/100 Test set: Average loss: 0.0096/13.9309, Accuracy: 50000/5733 (100.00%)/(57.3
3%) lr=0.1
59/100 Test set: Average loss: 0.0095/13.9643, Accuracy: 50000/5745 (100.00%)/(57.4
5%) lr=0.1
```

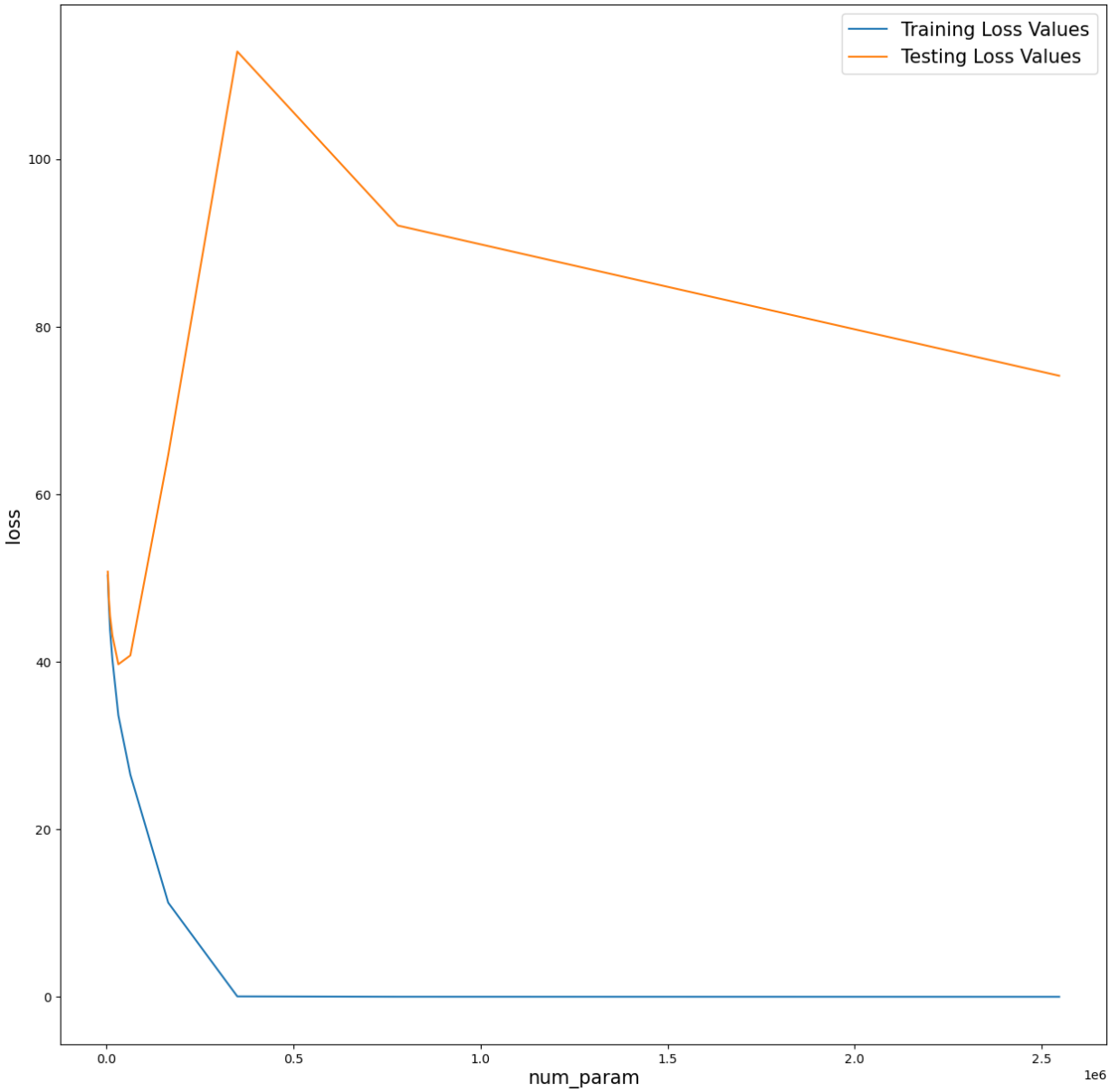
```
60/100 Test set: Average loss: 0.0092/14.0139, Accuracy: 50000/5739 (100.00%)/(57.3
9%) lr=0.1
61/100 Test set: Average loss: 0.0089/14.0314, Accuracy: 50000/5732 (100.00%)/(57.3
2%) lr=0.1
62/100 Test set: Average loss: 0.0087/14.0520, Accuracy: 50000/5723 (100.00%)/(57.2
3%) lr=0.1
63/100 Test set: Average loss: 0.0086/14.0755, Accuracy: 50000/5729 (100.00%)/(57.2
9%) lr=0.1
64/100 Test set: Average loss: 0.0084/14.1003, Accuracy: 50000/5723 (100.00%)/(57.2
3%) lr=0.1
65/100 Test set: Average loss: 0.0083/14.1332, Accuracy: 50000/5728 (100.00%)/(57.2
8%) lr=0.1
66/100 Test set: Average loss: 0.0081/14.1477, Accuracy: 50000/5726 (100.00%)/(57.2
6%) lr=0.1
67/100 Test set: Average loss: 0.0079/14.2010, Accuracy: 50000/5729 (100.00%)/(57.2
9%) lr=0.1
68/100 Test set: Average loss: 0.0078/14.2157, Accuracy: 50000/5729 (100.00%)/(57.2
9%) lr=0.1
69/100 Test set: Average loss: 0.0076/14.2440, Accuracy: 50000/5723 (100.00%)/(57.2
3%) lr=0.1
70/100 Test set: Average loss: 0.0074/14.2568, Accuracy: 50000/5726 (100.00%)/(57.2
6%) lr=0.1
71/100 Test set: Average loss: 0.0073/14.2848, Accuracy: 50000/5732 (100.00%)/(57.3
2%) lr=0.1
72/100 Test set: Average loss: 0.0071/14.3039, Accuracy: 50000/5732 (100.00%)/(57.3
2%) lr=0.1
73/100 Test set: Average loss: 0.0071/14.3341, Accuracy: 50000/5720 (100.00%)/(57.2
0%) lr=0.1
74/100 Test set: Average loss: 0.0069/14.3476, Accuracy: 50000/5723 (100.00%)/(57.2
3%) lr=0.1
75/100 Test set: Average loss: 0.0067/14.3768, Accuracy: 50000/5726 (100.00%)/(57.2
6%) lr=0.1
76/100 Test set: Average loss: 0.0067/14.4021, Accuracy: 50000/5728 (100.00%)/(57.2
8%) lr=0.1
77/100 Test set: Average loss: 0.0065/14.4210, Accuracy: 50000/5723 (100.00%)/(57.2
3%) lr=0.1
78/100 Test set: Average loss: 0.0065/14.4413, Accuracy: 50000/5730 (100.00%)/(57.3
0%) lr=0.1
79/100 Test set: Average loss: 0.0064/14.4781, Accuracy: 50000/5718 (100.00%)/(57.1
8%) lr=0.1
80/100 Test set: Average loss: 0.0062/14.4855, Accuracy: 50000/5728 (100.00%)/(57.2
8%) lr=0.1
81/100 Test set: Average loss: 0.0061/14.5007, Accuracy: 50000/5726 (100.00%)/(57.2
6%) lr=0.1
82/100 Test set: Average loss: 0.0061/14.5267, Accuracy: 50000/5736 (100.00%)/(57.3
6%) lr=0.1
83/100 Test set: Average loss: 0.0059/14.5442, Accuracy: 50000/5722 (100.00%)/(57.2
2%) lr=0.1
84/100 Test set: Average loss: 0.0059/14.5683, Accuracy: 50000/5725 (100.00%)/(57.2
5%) lr=0.1
85/100 Test set: Average loss: 0.0058/14.5878, Accuracy: 50000/5725 (100.00%)/(57.2
5%) lr=0.1
86/100 Test set: Average loss: 0.0056/14.5981, Accuracy: 50000/5731 (100.00%)/(57.3
1%) lr=0.1
87/100 Test set: Average loss: 0.0055/14.6224, Accuracy: 50000/5726 (100.00%)/(57.2
6%) lr=0.1
88/100 Test set: Average loss: 0.0055/14.6303, Accuracy: 50000/5723 (100.00%)/(57.2
3%) lr=0.1
89/100 Test set: Average loss: 0.0054/14.6566, Accuracy: 50000/5724 (100.00%)/(57.2
4%) lr=0.1
```

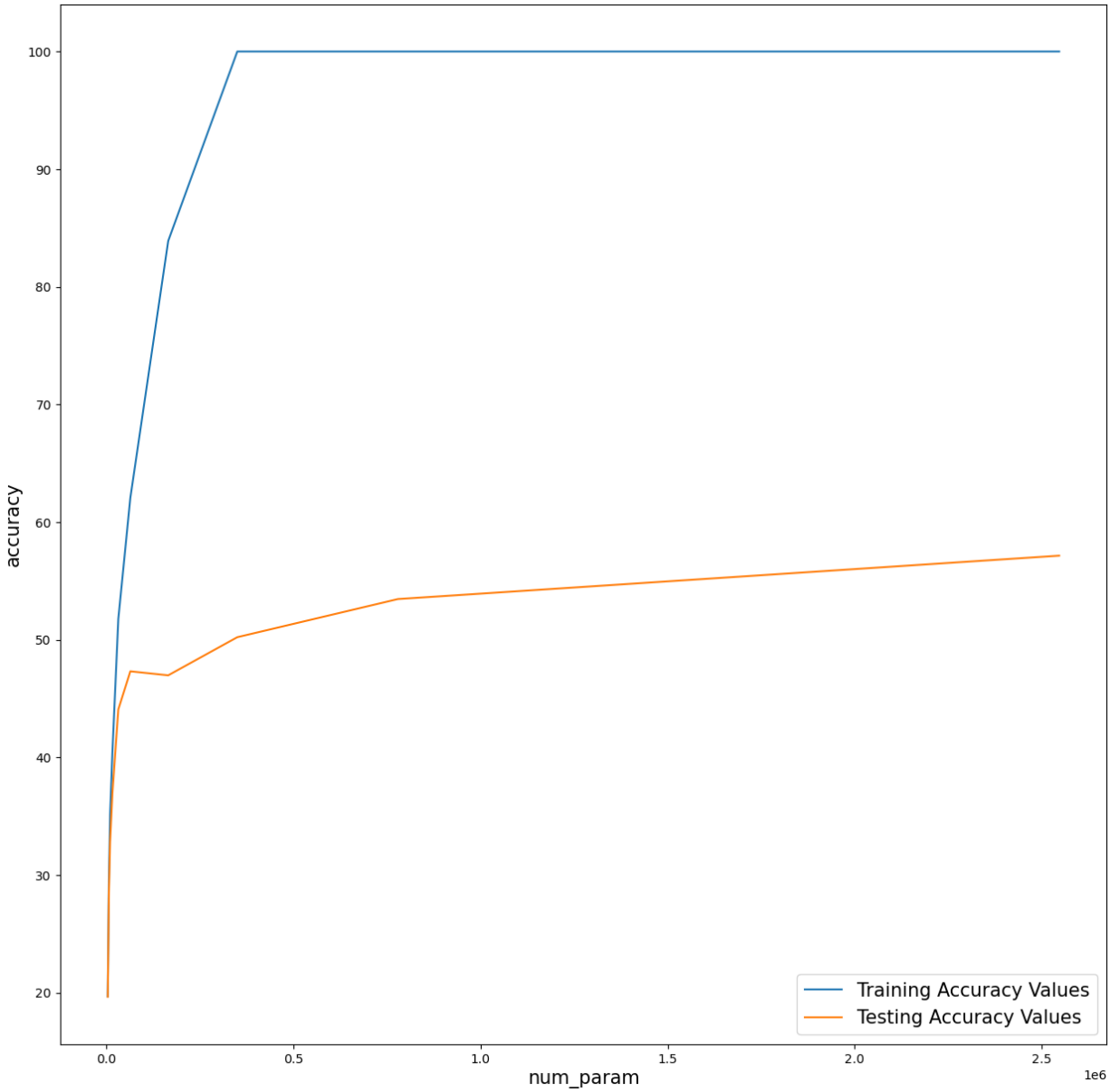
```
90/100 Test set: Average loss: 0.0054/14.6790, Accuracy: 50000/5720 (100.00%)/(57.2
0%) lr=0.1
91/100 Test set: Average loss: 0.0053/14.6966, Accuracy: 50000/5726 (100.00%)/(57.2
6%) lr=0.1
92/100 Test set: Average loss: 0.0052/14.7035, Accuracy: 50000/5713 (100.00%)/(57.1
3%) lr=0.1
93/100 Test set: Average loss: 0.0052/14.7250, Accuracy: 50000/5718 (100.00%)/(57.1
8%) lr=0.1
94/100 Test set: Average loss: 0.0051/14.7298, Accuracy: 50000/5717 (100.00%)/(57.1
7%) lr=0.1
95/100 Test set: Average loss: 0.0049/14.7634, Accuracy: 50000/5723 (100.00%)/(57.2
3%) lr=0.1
96/100 Test set: Average loss: 0.0049/14.7718, Accuracy: 50000/5718 (100.00%)/(57.1
8%) lr=0.1
97/100 Test set: Average loss: 0.0049/14.7916, Accuracy: 50000/5717 (100.00%)/(57.1
7%) lr=0.1
98/100 Test set: Average loss: 0.0048/14.8017, Accuracy: 50000/5722 (100.00%)/(57.2
2%) lr=0.1
99/100 Test set: Average loss: 0.0048/14.8273, Accuracy: 50000/5714 (100.00%)/(57.1
4%) lr=0.1
```

In [8]:
```
num_param_list = [num_param_1,num_param_2,num_param_3,num_param_4,num_param_5,num_para

trainloss_list = [trainloss_1,trainloss_2,trainloss_3,trainloss_4,trainloss_5,trainlos
trainloss = np.array(trainloss_list)[:,-1]
testloss_list = [testloss_1,testloss_2,testloss_3,testloss_4,testloss_5,testloss_6,tes
testloss = np.array(testloss_list)[:,-1]

train_acc_list = [train_acc_1,train_acc_2,train_acc_3,train_acc_4,train_acc_5,train_ac
train_acc = np.array(train_acc_list)[:,-1]
test_acc_list = [test_acc_1,test_acc_2,test_acc_3,test_acc_4,test_acc_5,test_acc_6,tes
test_acc = np.array(test_acc_list)[:,-1]
```

7. Ploting and Visualization

In [9]:
```
plt.figure(figsize=(15,15))
plt.plot((num_param_list), (trainloss), label='Training Loss Values')
plt.plot((num_param_list), (testloss*5), label='Testing Loss Values')
plt.xlabel('num_param',fontsize=15)
plt.ylabel('loss',fontsize=15)
plt.legend(fontsize=15)
plt.show()

plt.figure(figsize=(15,15))
plt.plot(num_param_list, train_acc, label='Training Accuracy Values')
plt.plot(num_param_list, test_acc, label='Testing Accuracy Values')
plt.xlabel('num_param',fontsize=15)
plt.ylabel('accuracy',fontsize=15)
plt.legend(fontsize=15)
plt.show()
```


Training Loss Values
Testing Loss Values
loss
num_param
1e6
0
20
40
60
80
100
0.0
0.5
1.0
1.5
2.0
2.5



8. Part1: Flatness v.s. Generalization

In [10]:
```
# 1. Training Function
def train_MNIST(model_name,
                Epochs = 20,
                Batch  = 2000,
                Data_workers = 0,
                LR = 0.1):

# 2. Initialization
    train_set = torchvision.datasets.MNIST(root='./data/',train=True,download=True,tra
    test_set = torchvision.datasets.MNIST(root='./data/',train=False,download=True,tra
    trainloader = DataLoader(train_set, batch_size=Batch, shuffle=True, num_workers=Da
    testloader  = DataLoader(test_set,  batch_size=Batch, shuffle=True, num_workers=Da
    print(train_set.classes)
    print(train_set.data.shape)
    print(test_set.data.shape)

# 3. Initialization Model
    torch.cuda.is_available()
```

```
    device = torch.device('cuda:0' if torch.cuda.is_available() else 'cpu')
    Model = model_name.to(device)

# 4. Loss & Optimizer
    criterion = nn.CrossEntropyLoss()
    optimizer = optim.SGD(Model.parameters(), lr=LR, momentum=0.9)
    scheduler = optim.lr_scheduler.StepLR(optimizer, step_size = 5, gamma = 0.8)

# 5. Training
    trainloss_list = []
    testloss_list  = []
    accuracy_list  = []
    lr_list = []
    F = []

    for epoch in range(Epochs):
        Model.train()
        train_loss = 0.0
        for i, data in enumerate(trainloader):
            images, labels = data
            images = (images.view(-1, 28*28)).to(device)
            labels = labels.to(device)
            outputs = Model(images)
            loss = criterion(outputs, labels)
            optimizer.zero_grad()
            loss.backward()
            optimizer.step()
            train_loss += loss.item()
            total = (i+1)*Batch

# 6. Evaluation
        Model.eval()
        with torch.no_grad():
            test_loss = 0
            correct = 0
            total = 0
            for data in testloader:
                images, labels = data
                images = (images.view(-1, 28*28)).to(device)
                labels = labels.to(device)
                outputs = Model(images)
                loss = criterion(outputs, labels)
                test_loss += loss.item()
                _, pred = torch.max(outputs.data, 1)
                correct += (pred == labels).cpu().sum()
                total += labels.size(0)
            total = len(testloader.dataset)
            accuracy = 100.0*correct/total

# 7. Save Loss
        lr_list.append(optimizer.state_dict()['param_groups'][0]['lr'])
        trainloss_list.append(train_loss)
        testloss_list.append(test_loss)
        accuracy_list.append(accuracy)
        print('{}/{} Test set: Average loss: {:.4f}/{:.4f}, Accuracy: {}/{} ({:.2f}%)
                epoch, Epochs, train_loss,test_loss, correct, total, accuracy, lr_list

    return [Model,
            trainloss_list,
            testloss_list,
```

```
            accuracy_list,
            lr_list,
            F]
```

8. Part1.1: Batch = 64 vs Batch = 2048

In [11]:
```
[Model_64,trainloss_64,testloss_64,accuracy_64,lr_64,_] = train_MNIST(model_name=DNN_M
[Model_2048,trainloss_2048,testloss_2048,accuracy_2048,lr_2048,_] = train_MNIST(model_
```

```
['0 - zero', '1 - one', '2 - two', '3 - three', '4 - four', '5 - five', '6 - six', '7
- seven', '8 - eight', '9 - nine']
torch.Size([60000, 28, 28])
torch.Size([10000, 28, 28])
0/20 Test set: Average loss: 207.0145/17.3169, Accuracy: 9687/10000 (96.87%) lr=0.1
1/20 Test set: Average loss: 100.5276/13.0820, Accuracy: 9765/10000 (97.65%) lr=0.1
2/20 Test set: Average loss: 74.3310/12.2023, Accuracy: 9751/10000 (97.51%) lr=0.1
3/20 Test set: Average loss: 60.2526/11.9346, Accuracy: 9776/10000 (97.76%) lr=0.1
4/20 Test set: Average loss: 49.3389/11.0850, Accuracy: 9794/10000 (97.94%) lr=0.1
5/20 Test set: Average loss: 42.0044/9.7915, Accuracy: 9806/10000 (98.06%) lr=0.1
6/20 Test set: Average loss: 36.4144/11.3653, Accuracy: 9792/10000 (97.92%) lr=0.1
7/20 Test set: Average loss: 30.9662/10.6907, Accuracy: 9800/10000 (98.00%) lr=0.1
8/20 Test set: Average loss: 28.4442/12.6325, Accuracy: 9781/10000 (97.81%) lr=0.1
9/20 Test set: Average loss: 26.2892/10.5030, Accuracy: 9811/10000 (98.11%) lr=0.1
10/20 Test set: Average loss: 23.3315/10.1873, Accuracy: 9802/10000 (98.02%) lr=0.1
11/20 Test set: Average loss: 22.4138/9.8865, Accuracy: 9828/10000 (98.28%) lr=0.1
12/20 Test set: Average loss: 18.9414/11.2778, Accuracy: 9800/10000 (98.00%) lr=0.1
13/20 Test set: Average loss: 15.5851/10.9160, Accuracy: 9813/10000 (98.13%) lr=0.1
14/20 Test set: Average loss: 16.6192/9.9364, Accuracy: 9829/10000 (98.29%) lr=0.1
15/20 Test set: Average loss: 17.2151/11.1163, Accuracy: 9822/10000 (98.22%) lr=0.1
16/20 Test set: Average loss: 13.0501/10.8873, Accuracy: 9826/10000 (98.26%) lr=0.1
17/20 Test set: Average loss: 14.0984/10.2755, Accuracy: 9831/10000 (98.31%) lr=0.1
18/20 Test set: Average loss: 11.8329/13.1605, Accuracy: 9801/10000 (98.01%) lr=0.1
19/20 Test set: Average loss: 12.6670/11.1155, Accuracy: 9824/10000 (98.24%) lr=0.1
['0 - zero', '1 - one', '2 - two', '3 - three', '4 - four', '5 - five', '6 - six', '7
- seven', '8 - eight', '9 - nine']
torch.Size([60000, 28, 28])
torch.Size([10000, 28, 28])
0/20 Test set: Average loss: 16.8588/1.0952, Accuracy: 9345/10000 (93.45%) lr=0.1
1/20 Test set: Average loss: 4.3265/0.6510, Accuracy: 9603/10000 (96.03%) lr=0.1
2/20 Test set: Average loss: 2.6766/0.5022, Accuracy: 9695/10000 (96.95%) lr=0.1
3/20 Test set: Average loss: 1.9545/0.4226, Accuracy: 9733/10000 (97.33%) lr=0.1
4/20 Test set: Average loss: 1.4775/0.3854, Accuracy: 9756/10000 (97.56%) lr=0.1
5/20 Test set: Average loss: 1.1221/0.3716, Accuracy: 9770/10000 (97.70%) lr=0.1
6/20 Test set: Average loss: 0.8268/0.3916, Accuracy: 9772/10000 (97.72%) lr=0.1
7/20 Test set: Average loss: 0.6566/0.3457, Accuracy: 9789/10000 (97.89%) lr=0.1
8/20 Test set: Average loss: 0.4894/0.3368, Accuracy: 9793/10000 (97.93%) lr=0.1
9/20 Test set: Average loss: 0.3694/0.3588, Accuracy: 9792/10000 (97.92%) lr=0.1
10/20 Test set: Average loss: 0.3029/0.3497, Accuracy: 9787/10000 (97.87%) lr=0.1
11/20 Test set: Average loss: 0.2289/0.3532, Accuracy: 9787/10000 (97.87%) lr=0.1
12/20 Test set: Average loss: 0.1698/0.3574, Accuracy: 9791/10000 (97.91%) lr=0.1
13/20 Test set: Average loss: 0.1443/0.3590, Accuracy: 9791/10000 (97.91%) lr=0.1
14/20 Test set: Average loss: 0.1099/0.3604, Accuracy: 9796/10000 (97.96%) lr=0.1
15/20 Test set: Average loss: 0.0797/0.3557, Accuracy: 9799/10000 (97.99%) lr=0.1
16/20 Test set: Average loss: 0.0658/0.3689, Accuracy: 9795/10000 (97.95%) lr=0.1
17/20 Test set: Average loss: 0.0568/0.3672, Accuracy: 9795/10000 (97.95%) lr=0.1
18/20 Test set: Average loss: 0.0499/0.3705, Accuracy: 9791/10000 (97.91%) lr=0.1
19/20 Test set: Average loss: 0.0484/0.3668, Accuracy: 9803/10000 (98.03%) lr=0.1
```

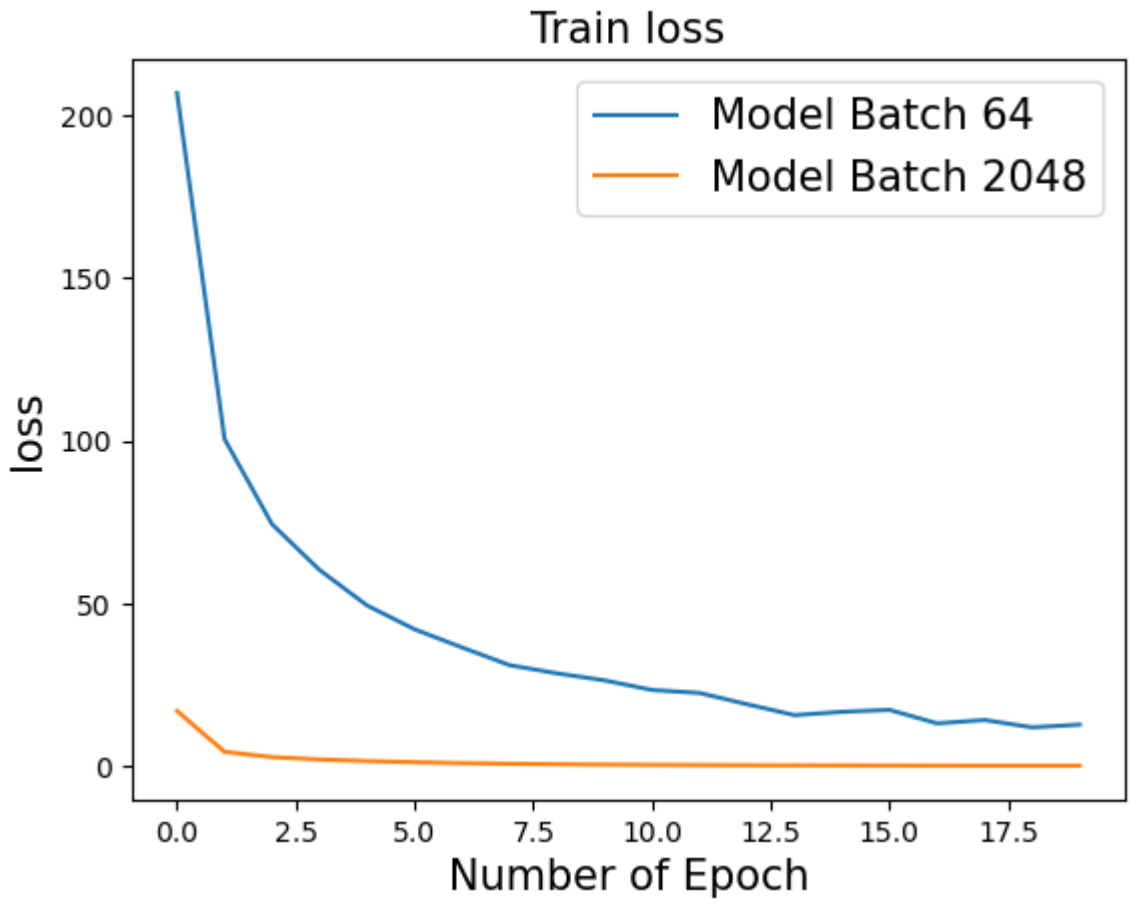
8. Part1.2: Learning Rate = 0.01 vs Learning Rate = 0.001

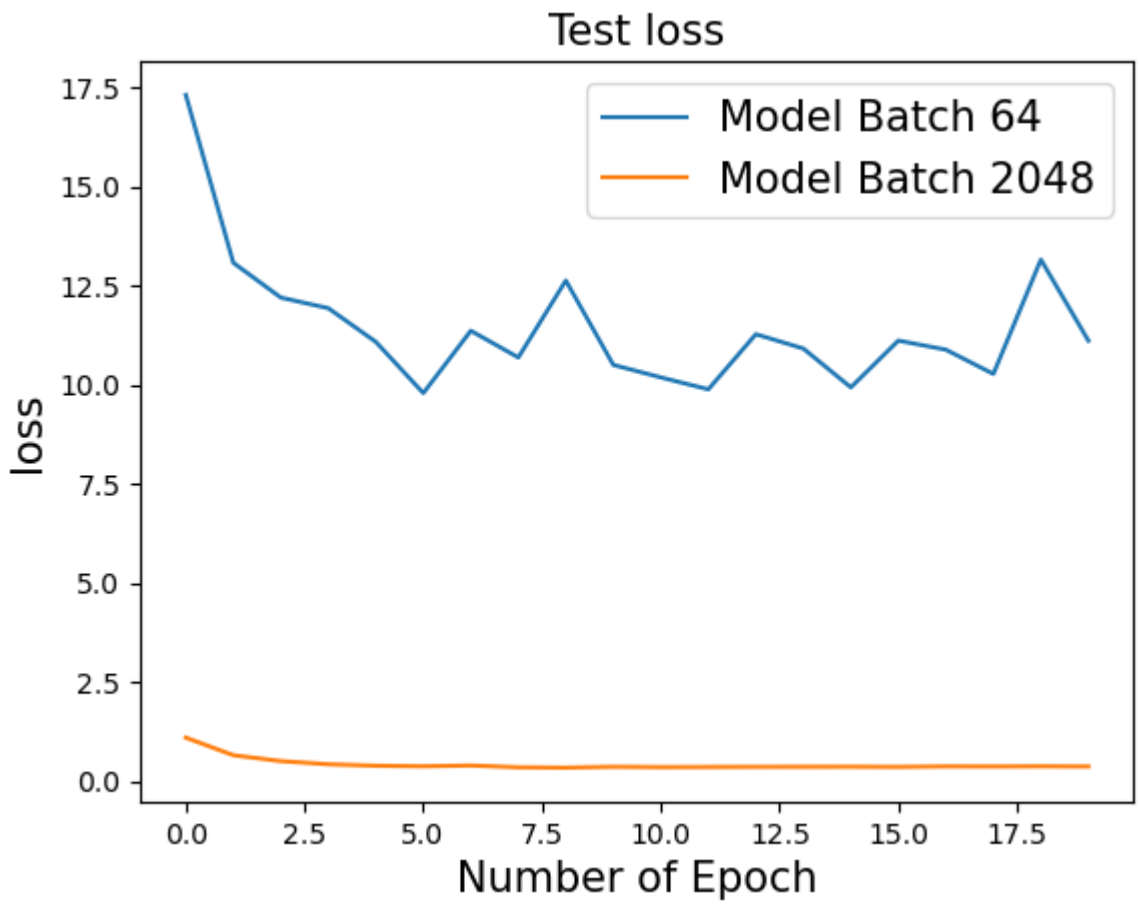
In [12]:
```
[Model_1e3,trainloss_1e3,testloss_1e3,accuracy_1e3,lr_1e3,_] = train_MNIST(model_name=
[Model_1e2,trainloss_1e2,testloss_1e2,accuracy_1e2,lr_1e2,_] = train_MNIST(model_name=
```

```
['0 - zero', '1 - one', '2 - two', '3 - three', '4 - four', '5 - five', '6 - six', '7
- seven', '8 - eight', '9 - nine']
torch.Size([60000, 28, 28])
torch.Size([10000, 28, 28])
0/20 Test set: Average loss: 62.6735/9.8471, Accuracy: 5111/10000 (51.11%) lr=0.001
1/20 Test set: Average loss: 43.4908/6.1466, Accuracy: 7794/10000 (77.94%) lr=0.001
2/20 Test set: Average loss: 33.2672/4.7786, Accuracy: 8340/10000 (83.40%) lr=0.001
3/20 Test set: Average loss: 26.7201/3.9075, Accuracy: 8630/10000 (86.30%) lr=0.001
4/20 Test set: Average loss: 22.1768/3.2979, Accuracy: 8820/10000 (88.20%) lr=0.001
5/20 Test set: Average loss: 18.9577/2.8582, Accuracy: 8931/10000 (89.31%) lr=0.001
6/20 Test set: Average loss: 16.6248/2.5293, Accuracy: 9035/10000 (90.35%) lr=0.001
7/20 Test set: Average loss: 14.8586/2.2837, Accuracy: 9097/10000 (90.97%) lr=0.001
8/20 Test set: Average loss: 13.4973/2.0893, Accuracy: 9145/10000 (91.45%) lr=0.001
9/20 Test set: Average loss: 12.4102/1.9289, Accuracy: 9186/10000 (91.86%) lr=0.001
10/20 Test set: Average loss: 11.5161/1.8003, Accuracy: 9222/10000 (92.22%) lr=0.001
11/20 Test set: Average loss: 10.7770/1.6919, Accuracy: 9251/10000 (92.51%) lr=0.001
12/20 Test set: Average loss: 10.1513/1.6026, Accuracy: 9278/10000 (92.78%) lr=0.001
13/20 Test set: Average loss: 9.6023/1.5189, Accuracy: 9312/10000 (93.12%) lr=0.001
14/20 Test set: Average loss: 9.1178/1.4515, Accuracy: 9339/10000 (93.39%) lr=0.001
15/20 Test set: Average loss: 8.7024/1.3882, Accuracy: 9366/10000 (93.66%) lr=0.001
16/20 Test set: Average loss: 8.3271/1.3344, Accuracy: 9385/10000 (93.85%) lr=0.001
17/20 Test set: Average loss: 7.9845/1.2830, Accuracy: 9397/10000 (93.97%) lr=0.001
18/20 Test set: Average loss: 7.6809/1.2372, Accuracy: 9416/10000 (94.16%) lr=0.001
19/20 Test set: Average loss: 7.3951/1.1959, Accuracy: 9441/10000 (94.41%) lr=0.001
['0 - zero', '1 - one', '2 - two', '3 - three', '4 - four', '5 - five', '6 - six', '7
- seven', '8 - eight', '9 - nine']
torch.Size([60000, 28, 28])
torch.Size([10000, 28, 28])
0/20 Test set: Average loss: 38.4621/4.2540, Accuracy: 8773/10000 (87.73%) lr=0.01
1/20 Test set: Average loss: 12.1265/1.5019, Accuracy: 9268/10000 (92.68%) lr=0.01
2/20 Test set: Average loss: 7.5520/1.0828, Accuracy: 9429/10000 (94.29%) lr=0.01
3/20 Test set: Average loss: 5.8438/0.8994, Accuracy: 9515/10000 (95.15%) lr=0.01
4/20 Test set: Average loss: 4.8450/0.7787, Accuracy: 9569/10000 (95.69%) lr=0.01
5/20 Test set: Average loss: 4.1624/0.6996, Accuracy: 9628/10000 (96.28%) lr=0.01
6/20 Test set: Average loss: 3.6477/0.6441, Accuracy: 9640/10000 (96.40%) lr=0.01
7/20 Test set: Average loss: 3.2276/0.5931, Accuracy: 9663/10000 (96.63%) lr=0.01
8/20 Test set: Average loss: 2.8893/0.5591, Accuracy: 9678/10000 (96.78%) lr=0.01
9/20 Test set: Average loss: 2.6049/0.5310, Accuracy: 9687/10000 (96.87%) lr=0.01
10/20 Test set: Average loss: 2.3542/0.5086, Accuracy: 9712/10000 (97.12%) lr=0.01
11/20 Test set: Average loss: 2.1410/0.4846, Accuracy: 9725/10000 (97.25%) lr=0.01
12/20 Test set: Average loss: 1.9552/0.4710, Accuracy: 9723/10000 (97.23%) lr=0.01
13/20 Test set: Average loss: 1.7954/0.4500, Accuracy: 9729/10000 (97.29%) lr=0.01
14/20 Test set: Average loss: 1.6393/0.4386, Accuracy: 9743/10000 (97.43%) lr=0.01
15/20 Test set: Average loss: 1.5085/0.4279, Accuracy: 9754/10000 (97.54%) lr=0.01
16/20 Test set: Average loss: 1.3965/0.4177, Accuracy: 9761/10000 (97.61%) lr=0.01
17/20 Test set: Average loss: 1.2773/0.4060, Accuracy: 9760/10000 (97.60%) lr=0.01
18/20 Test set: Average loss: 1.1903/0.4110, Accuracy: 9752/10000 (97.52%) lr=0.01
19/20 Test set: Average loss: 1.1038/0.4064, Accuracy: 9762/10000 (97.62%) lr=0.01
```

In [13]:
```
plt.figure()
plt.plot(trainloss_64, label='Model Batch 64')
plt.plot(trainloss_2048, label='Model Batch 2048')
plt.xlabel('Number of Epoch',fontsize=15)
```

```
plt.ylabel('loss',fontsize=15)
plt.title('Train loss',fontsize=15)
plt.legend(fontsize=15)
plt.show()

plt.figure()
plt.plot(testloss_64, label='Model Batch 64')
plt.plot(testloss_2048, label='Model Batch 2048')
plt.xlabel('Number of Epoch',fontsize=15)
plt.ylabel('loss',fontsize=15)
plt.title('Test loss',fontsize=15)
plt.legend(fontsize=15)
plt.show()
```





In [14]:
```
train_set = torchvision.datasets.MNIST(root='./data/',train=True,download=True,transfo
test_set = torchvision.datasets.MNIST(root='./data/',train=False,download=True,transfo
trainloader = DataLoader(train_set, batch_size=2000, shuffle=True, num_workers=0)
testloader  = DataLoader(test_set,  batch_size=2000, shuffle=True, num_workers=0)
device = torch.device('cuda:0' if torch.cuda.is_available() else 'cpu')
Model = DNN_MNIST_N(28*28,100,200,100,10).to(device)
criterion = nn.CrossEntropyLoss()
param_1 = Model_64.state_dict()
param_2 = Model_2048.state_dict()

batch_train_loss = []
batch_test_loss = []
batch_train_acc = []
batch_test_acc = []

alpha_list = np.linspace(-2,2,50)
for i in range(len(alpha_list)):
    alpha = alpha_list[i]
    param_new = {}
    for key in param_1.keys():
        param_new[key] = (1-alpha)*param_1[key] + alpha*param_2[key]
    Model.load_state_dict(param_new)
    Model.eval()
    with torch.no_grad():
        train_loss = 0
        correct = 0
        for data in trainloader:
            images, labels = data
            images = (images.view(-1, 28*28)).to(device)
            labels = labels.to(device)
```

```
            outputs = Model(images)
            loss = criterion(outputs, labels)
            train_loss += loss.item()
            _, pred = torch.max(outputs.data, 1)
            correct += (pred == labels).cpu().sum()
        total = len(trainloader.dataset)
        accuracy = 100.0*correct/total
        batch_train_loss.append(loss.detach().cpu().numpy())
        batch_train_acc.append(accuracy.detach().cpu().numpy())

        test_loss = 0
        correct = 0
        for data in testloader:
            images, labels = data
            images = (images.view(-1, 28*28)).to(device)
            labels = labels.to(device)
            outputs = Model(images)
            loss = criterion(outputs, labels)
            test_loss += loss.item()
            _, pred = torch.max(outputs.data, 1)
            correct += (pred == labels).cpu().sum()
        total = len(testloader.dataset)
        accuracy = 100.0*correct/total
        batch_test_loss.append(loss.detach().cpu().numpy())
        batch_test_acc.append(accuracy.detach().cpu().numpy())

    print(alpha)
```

```
-2.0
-1.9183673469387754
-1.836734693877551
-1.7551020408163265
-1.6734693877551021
-1.5918367346938775
-1.510204081632653
-1.4285714285714286
-1.3469387755102042
-1.2653061224489797
-1.183673469387755
-1.1020408163265307
-1.0204081632653061
-0.9387755102040818
-0.8571428571428572
-0.7755102040816328
-0.6938775510204083
-0.6122448979591837
-0.5306122448979593
-0.44897959183673475
-0.3673469387755104
-0.2857142857142858
-0.20408163265306145
-0.12244897959183687
-0.04081632653061229
0.04081632653061229
0.12244897959183643
0.204081632653061
0.2857142857142856
0.36734693877551017
0.4489795918367343
0.5306122448979589
0.6122448979591835
0.693877551020408
0.7755102040816326
0.8571428571428568
0.9387755102040813
1.020408163265306
1.1020408163265305
1.1836734693877546
1.2653061224489792
1.3469387755102038
1.4285714285714284
1.510204081632653
1.591836734693877
1.6734693877551017
1.7551020408163263
1.8367346938775508
1.9183673469387754
2.0
```

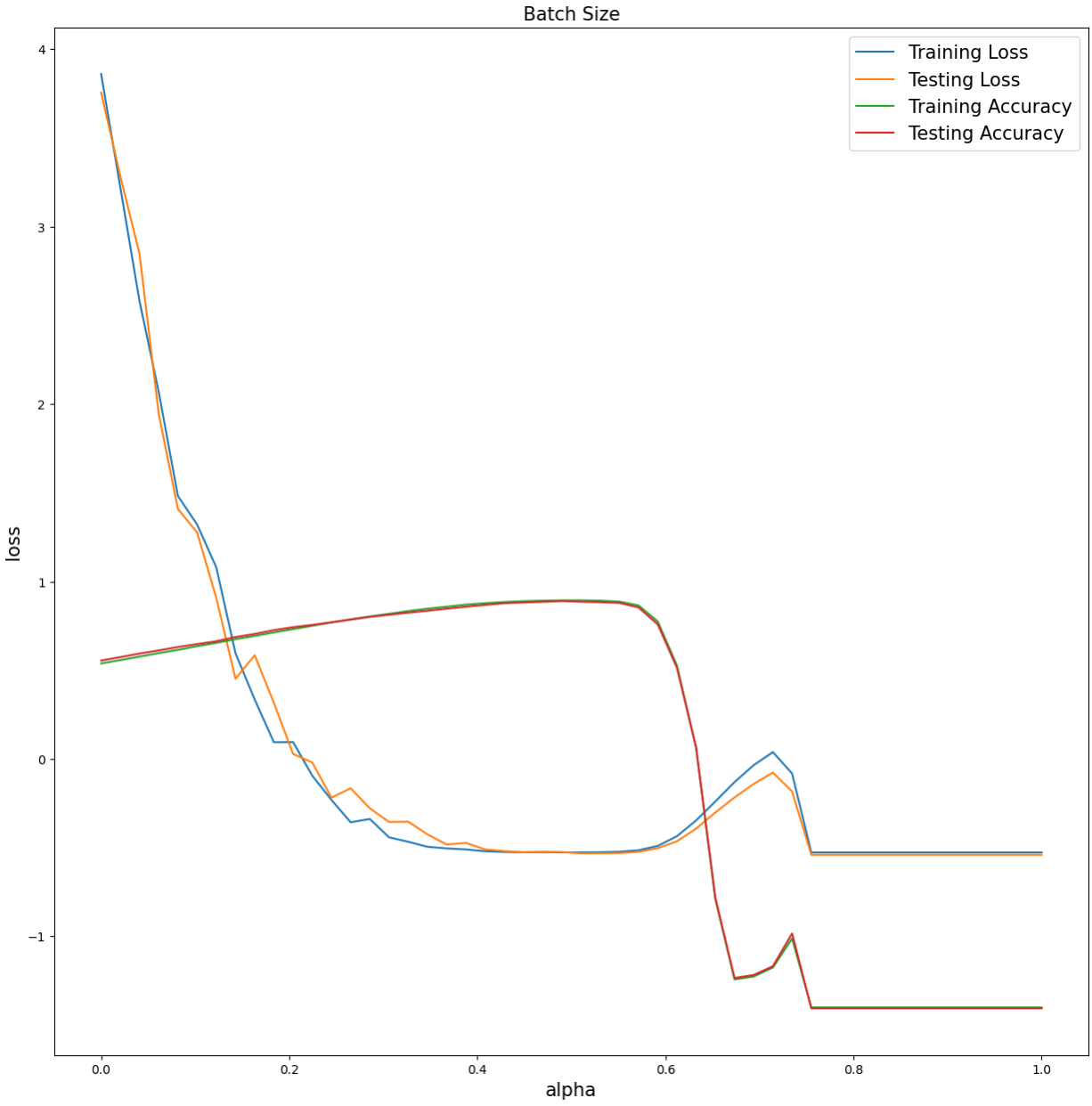
In [15]:
```
train_set = torchvision.datasets.MNIST(root='./data/',train=True,download=True,transfo
test_set = torchvision.datasets.MNIST(root='./data/',train=False,download=True,transfo
trainloader = DataLoader(train_set, batch_size=2000, shuffle=True, num_workers=0)
testloader  = DataLoader(test_set,  batch_size=2000, shuffle=True, num_workers=0)
device = torch.device('cuda:0' if torch.cuda.is_available() else 'cpu')
Model = DNN_MNIST_N(28*28,100,200,100,10).to(device)
criterion = nn.CrossEntropyLoss()
param_1 = Model_1e3.state_dict()
param_2 = Model_1e2.state_dict()
```

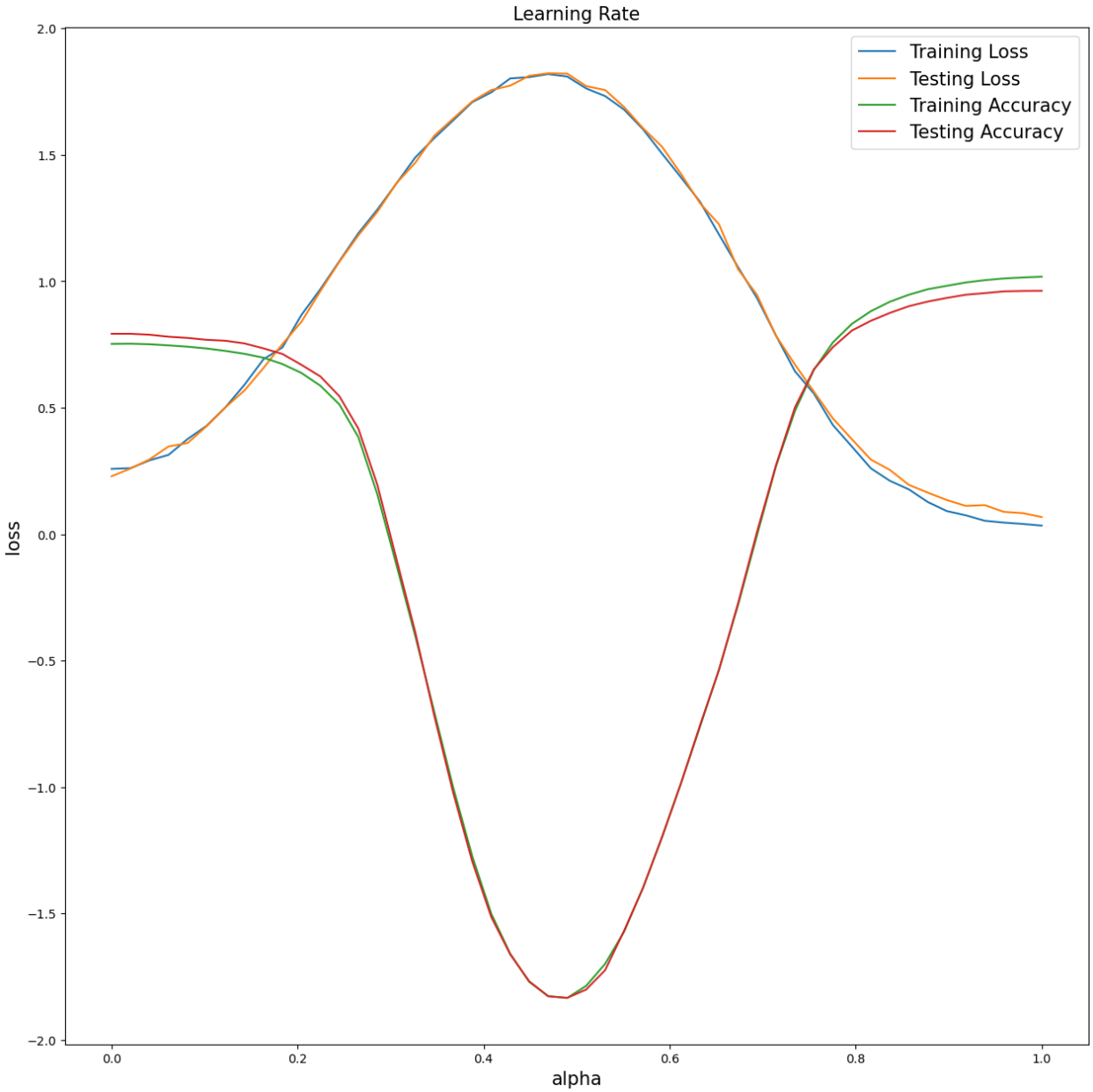
```
lr_train_loss = []
lr_train_acc = []
lr_test_loss = []
lr_test_acc = []

alpha_list = np.linspace(0,1,50)
for i in range(len(alpha_list)):
    alpha = alpha_list[i]
    param_new = {}
    for key in param_1.keys():
        param_new[key] = (1-alpha)*param_1[key] + alpha*param_2[key]
    Model.load_state_dict(param_new)
    Model.eval()
    with torch.no_grad():
        train_loss = 0
        correct = 0
        for data in trainloader:
            images, labels = data
            images = (images.view(-1, 28*28)).to(device)
            labels = labels.to(device)
            outputs = Model(images)
            loss = criterion(outputs, labels)
            train_loss += loss.item()
            _, pred = torch.max(outputs.data, 1)
            correct += (pred == labels).cpu().sum()
        total = len(trainloader.dataset)
        accuracy = 100.0*correct/total
        lr_train_loss.append(loss.detach().cpu().numpy())
        lr_train_acc.append(accuracy.detach().cpu().numpy())

        test_loss = 0
        correct = 0
        for data in testloader:
            images, labels = data
            images = (images.view(-1, 28*28)).to(device)
            labels = labels.to(device)
            outputs = Model(images)
            loss = criterion(outputs, labels)
            test_loss += loss.item()
            _, pred = torch.max(outputs.data, 1)
            correct += (pred == labels).cpu().sum()
        total = len(testloader.dataset)
        accuracy = 100.0*correct/total
        lr_test_loss.append(loss.detach().cpu().numpy())
        lr_test_acc.append(accuracy.detach().cpu().numpy())

    print(alpha)
```

```
0.0
0.02040816326530612
0.04081632653061224
0.061224489795918366
0.08163265306122448
0.1020408163265306
0.12244897959183673
0.14285714285714285
0.16326530612244897
0.18367346938775508
0.2040816326530612
0.22448979591836732
0.24489795918367346
0.26530612244897955
0.2857142857142857
0.3061224489795918
0.32653061224489793
0.3469387755102041
0.36734693877551017
0.3877551020408163
0.4081632653061224
0.42857142857142855
0.44897959183673464
0.4693877551020408
0.4897959183673469
0.5102040816326531
0.5306122448979591
0.5510204081632653
0.5714285714285714
0.5918367346938775
0.6122448979591836
0.6326530612244897
0.6530612244897959
0.673469387755102
0.6938775510204082
0.7142857142857142
0.7346938775510203
0.7551020408163265
0.7755102040816326
0.7959183673469387
0.8163265306122448
0.836734693877551
0.8571428571428571
0.8775510204081632
0.8979591836734693
0.9183673469387754
0.9387755102040816
0.9591836734693877
0.9795918367346939
1.0
```

In [16]:
```
plt.figure(figsize=(15,15))
plt.plot(alpha_list,standardization(batch_train_loss), label='Training Loss')
plt.plot(alpha_list,standardization(batch_test_loss),label='Testing Loss')
plt.plot(alpha_list,standardization(batch_train_acc),label='Training Accuracy')
plt.plot(alpha_list,standardization(batch_test_acc),label='Testing Accuracy')
plt.xlabel('alpha',fontsize=15)
plt.ylabel('loss',fontsize=15)
plt.title('Batch Size',fontsize=15)
plt.legend(fontsize=15)
```

```
plt.show()

plt.figure(figsize=(15,15))
plt.plot(alpha_list,(lr_train_loss), label='Training Loss')
plt.plot(alpha_list,(lr_test_loss), label='Testing Loss')
plt.plot(alpha_list,standardization(lr_train_acc), label='Training Accuracy')
plt.plot(alpha_list,standardization(lr_test_acc), label='Testing Accuracy')
plt.xlabel('alpha',fontsize=15)
plt.ylabel('loss',fontsize=15)
plt.title('Learning Rate',fontsize=15)
plt.legend(fontsize=15)
plt.show()
```





9. Part 2: Flatness vs Generalization

In [17]:
```
[_,trainloss_1,testloss_1,accuracy_1,_,F_1] = train_MNIST(model_name=DNN_MNIST_N(28*28
[_,trainloss_2,testloss_2,accuracy_2,_,F_2] = train_MNIST(model_name=DNN_MNIST_N(28*28
[_,trainloss_3,testloss_3,accuracy_3,_,F_3] = train_MNIST(model_name=DNN_MNIST_N(28*28
[_,trainloss_4,testloss_4,accuracy_4,_,F_4] = train_MNIST(model_name=DNN_MNIST_N(28*28
[_,trainloss_5,testloss_5,accuracy_5,_,F_5] = train_MNIST(model_name=DNN_MNIST_N(28*28
```

```
['0 - zero', '1 - one', '2 - two', '3 - three', '4 - four', '5 - five', '6 - six', '7
- seven', '8 - eight', '9 - nine']
torch.Size([60000, 28, 28])
torch.Size([10000, 28, 28])
0/20 Test set: Average loss: 5140.4750/508.4624, Accuracy: 8474/10000 (84.74%) lr=0.1
1/20 Test set: Average loss: 4340.8488/451.5248, Accuracy: 8601/10000 (86.01%) lr=0.1
2/20 Test set: Average loss: 4135.8618/353.4265, Accuracy: 8985/10000 (89.85%) lr=0.1
3/20 Test set: Average loss: 3966.5687/428.2202, Accuracy: 8613/10000 (86.13%) lr=0.1
4/20 Test set: Average loss: 3837.6247/444.7015, Accuracy: 8723/10000 (87.23%) lr=0.1
5/20 Test set: Average loss: 3796.7487/322.4521, Accuracy: 9001/10000 (90.01%) lr=0.1
6/20 Test set: Average loss: 3744.8574/379.4729, Accuracy: 8925/10000 (89.25%) lr=0.1
7/20 Test set: Average loss: 3725.1226/367.8878, Accuracy: 8907/10000 (89.07%) lr=0.1
8/20 Test set: Average loss: 3687.3285/381.7657, Accuracy: 8844/10000 (88.44%) lr=0.1
9/20 Test set: Average loss: 3586.9107/445.6295, Accuracy: 8533/10000 (85.33%) lr=0.1
10/20 Test set: Average loss: 3659.1679/428.2776, Accuracy: 8646/10000 (86.46%) lr=0.
1
11/20 Test set: Average loss: 3648.4746/345.0545, Accuracy: 9025/10000 (90.25%) lr=0.
1
12/20 Test set: Average loss: 3590.5883/311.8794, Accuracy: 9086/10000 (90.86%) lr=0.
1
13/20 Test set: Average loss: 3562.7432/405.0132, Accuracy: 8770/10000 (87.70%) lr=0.
1
14/20 Test set: Average loss: 3575.6323/337.6481, Accuracy: 9012/10000 (90.12%) lr=0.
1
15/20 Test set: Average loss: 3558.5076/355.6840, Accuracy: 9009/10000 (90.09%) lr=0.
1
16/20 Test set: Average loss: 3503.2898/299.5805, Accuracy: 9167/10000 (91.67%) lr=0.
1
17/20 Test set: Average loss: 3510.2591/326.6483, Accuracy: 9006/10000 (90.06%) lr=0.
1
18/20 Test set: Average loss: 3561.9472/401.1847, Accuracy: 8844/10000 (88.44%) lr=0.
1
19/20 Test set: Average loss: 3536.4898/286.0670, Accuracy: 9146/10000 (91.46%) lr=0.
1
['0 - zero', '1 - one', '2 - two', '3 - three', '4 - four', '5 - five', '6 - six', '7
- seven', '8 - eight', '9 - nine']
torch.Size([60000, 28, 28])
torch.Size([10000, 28, 28])
0/20 Test set: Average loss: 522.7066/49.9885, Accuracy: 9251/10000 (92.51%) lr=0.1
1/20 Test set: Average loss: 343.7774/44.6305, Accuracy: 9298/10000 (92.98%) lr=0.1
2/20 Test set: Average loss: 310.0862/40.7814, Accuracy: 9374/10000 (93.74%) lr=0.1
3/20 Test set: Average loss: 287.5375/38.6117, Accuracy: 9432/10000 (94.32%) lr=0.1
4/20 Test set: Average loss: 278.2798/38.0333, Accuracy: 9438/10000 (94.38%) lr=0.1
5/20 Test set: Average loss: 277.4725/40.1453, Accuracy: 9385/10000 (93.85%) lr=0.1
6/20 Test set: Average loss: 267.5490/37.9644, Accuracy: 9436/10000 (94.36%) lr=0.1
7/20 Test set: Average loss: 262.0336/38.2550, Accuracy: 9433/10000 (94.33%) lr=0.1
8/20 Test set: Average loss: 253.4466/41.0037, Accuracy: 9380/10000 (93.80%) lr=0.1
9/20 Test set: Average loss: 253.6755/36.4519, Accuracy: 9449/10000 (94.49%) lr=0.1
10/20 Test set: Average loss: 249.3305/35.4426, Accuracy: 9457/10000 (94.57%) lr=0.1
11/20 Test set: Average loss: 245.4422/35.8604, Accuracy: 9457/10000 (94.57%) lr=0.1
12/20 Test set: Average loss: 240.4014/35.9184, Accuracy: 9459/10000 (94.59%) lr=0.1
13/20 Test set: Average loss: 235.9402/36.1064, Accuracy: 9457/10000 (94.57%) lr=0.1
14/20 Test set: Average loss: 238.5654/34.8471, Accuracy: 9474/10000 (94.74%) lr=0.1
15/20 Test set: Average loss: 234.5882/33.4248, Accuracy: 9491/10000 (94.91%) lr=0.1
16/20 Test set: Average loss: 230.7624/34.3172, Accuracy: 9476/10000 (94.76%) lr=0.1
17/20 Test set: Average loss: 228.9706/33.6471, Accuracy: 9492/10000 (94.92%) lr=0.1
18/20 Test set: Average loss: 230.0397/34.8953, Accuracy: 9474/10000 (94.74%) lr=0.1
19/20 Test set: Average loss: 225.8205/32.8713, Accuracy: 9504/10000 (95.04%) lr=0.1
['0 - zero', '1 - one', '2 - two', '3 - three', '4 - four', '5 - five', '6 - six', '7
- seven', '8 - eight', '9 - nine']
```

```
torch.Size([60000, 28, 28])
torch.Size([10000, 28, 28])
0/20 Test set: Average loss: 253.7147/25.2554, Accuracy: 9225/10000 (92.25%) lr=0.1
1/20 Test set: Average loss: 152.6507/21.2692, Accuracy: 9374/10000 (93.74%) lr=0.1
2/20 Test set: Average loss: 138.8363/19.9617, Accuracy: 9400/10000 (94.00%) lr=0.1
3/20 Test set: Average loss: 126.0898/18.7646, Accuracy: 9461/10000 (94.61%) lr=0.1
4/20 Test set: Average loss: 119.3572/19.4090, Accuracy: 9430/10000 (94.30%) lr=0.1
5/20 Test set: Average loss: 115.4629/17.6272, Accuracy: 9473/10000 (94.73%) lr=0.1
6/20 Test set: Average loss: 111.0786/16.9201, Accuracy: 9498/10000 (94.98%) lr=0.1
7/20 Test set: Average loss: 107.0677/17.3393, Accuracy: 9479/10000 (94.79%) lr=0.1
8/20 Test set: Average loss: 105.2320/16.8552, Accuracy: 9510/10000 (95.10%) lr=0.1
9/20 Test set: Average loss: 102.6103/16.8098, Accuracy: 9517/10000 (95.17%) lr=0.1
10/20 Test set: Average loss: 100.6797/16.5129, Accuracy: 9523/10000 (95.23%) lr=0.1
11/20 Test set: Average loss: 98.3114/16.0536, Accuracy: 9539/10000 (95.39%) lr=0.1
12/20 Test set: Average loss: 97.3181/16.6356, Accuracy: 9520/10000 (95.20%) lr=0.1
13/20 Test set: Average loss: 94.9786/16.8530, Accuracy: 9515/10000 (95.15%) lr=0.1
14/20 Test set: Average loss: 95.0423/15.6405, Accuracy: 9569/10000 (95.69%) lr=0.1
15/20 Test set: Average loss: 94.4841/15.7421, Accuracy: 9531/10000 (95.31%) lr=0.1
16/20 Test set: Average loss: 94.0393/15.9498, Accuracy: 9543/10000 (95.43%) lr=0.1
17/20 Test set: Average loss: 94.3453/15.6281, Accuracy: 9544/10000 (95.44%) lr=0.1
18/20 Test set: Average loss: 92.7795/15.9324, Accuracy: 9540/10000 (95.40%) lr=0.1
19/20 Test set: Average loss: 91.0022/16.0740, Accuracy: 9546/10000 (95.46%) lr=0.1
['0 - zero', '1 - one', '2 - two', '3 - three', '4 - four', '5 - five', '6 - six', '7
- seven', '8 - eight', '9 - nine']
torch.Size([60000, 28, 28])
torch.Size([10000, 28, 28])
0/20 Test set: Average loss: 142.9398/12.6225, Accuracy: 9270/10000 (92.70%) lr=0.1
1/20 Test set: Average loss: 70.5798/9.8418, Accuracy: 9410/10000 (94.10%) lr=0.1
2/20 Test set: Average loss: 61.8888/9.7483, Accuracy: 9396/10000 (93.96%) lr=0.1
3/20 Test set: Average loss: 58.9941/9.7956, Accuracy: 9409/10000 (94.09%) lr=0.1
4/20 Test set: Average loss: 55.0476/9.3020, Accuracy: 9410/10000 (94.10%) lr=0.1
5/20 Test set: Average loss: 53.6560/8.8595, Accuracy: 9471/10000 (94.71%) lr=0.1
6/20 Test set: Average loss: 52.2759/8.6582, Accuracy: 9487/10000 (94.87%) lr=0.1
7/20 Test set: Average loss: 50.1870/9.1695, Accuracy: 9434/10000 (94.34%) lr=0.1
8/20 Test set: Average loss: 48.9749/8.4211, Accuracy: 9495/10000 (94.95%) lr=0.1
9/20 Test set: Average loss: 47.9387/8.4007, Accuracy: 9487/10000 (94.87%) lr=0.1
10/20 Test set: Average loss: 47.0718/8.3479, Accuracy: 9497/10000 (94.97%) lr=0.1
11/20 Test set: Average loss: 46.8086/8.3387, Accuracy: 9491/10000 (94.91%) lr=0.1
12/20 Test set: Average loss: 45.9979/8.3165, Accuracy: 9498/10000 (94.98%) lr=0.1
13/20 Test set: Average loss: 45.3763/8.7601, Accuracy: 9461/10000 (94.61%) lr=0.1
14/20 Test set: Average loss: 44.1589/8.7781, Accuracy: 9453/10000 (94.53%) lr=0.1
15/20 Test set: Average loss: 43.7787/8.3569, Accuracy: 9495/10000 (94.95%) lr=0.1
16/20 Test set: Average loss: 43.4553/8.5259, Accuracy: 9484/10000 (94.84%) lr=0.1
17/20 Test set: Average loss: 42.5515/8.1141, Accuracy: 9519/10000 (95.19%) lr=0.1
18/20 Test set: Average loss: 42.3363/8.2787, Accuracy: 9515/10000 (95.15%) lr=0.1
19/20 Test set: Average loss: 42.2970/8.5239, Accuracy: 9504/10000 (95.04%) lr=0.1
['0 - zero', '1 - one', '2 - two', '3 - three', '4 - four', '5 - five', '6 - six', '7
- seven', '8 - eight', '9 - nine']
torch.Size([60000, 28, 28])
torch.Size([10000, 28, 28])
0/20 Test set: Average loss: 53.4951/3.5763, Accuracy: 9001/10000 (90.01%) lr=0.1
1/20 Test set: Average loss: 16.9190/2.4831, Accuracy: 9245/10000 (92.45%) lr=0.1
2/20 Test set: Average loss: 13.7037/2.1893, Accuracy: 9353/10000 (93.53%) lr=0.1
3/20 Test set: Average loss: 12.2478/2.0373, Accuracy: 9388/10000 (93.88%) lr=0.1
4/20 Test set: Average loss: 11.3861/1.9899, Accuracy: 9400/10000 (94.00%) lr=0.1
5/20 Test set: Average loss: 10.6236/1.9685, Accuracy: 9412/10000 (94.12%) lr=0.1
6/20 Test set: Average loss: 10.2576/1.8888, Accuracy: 9437/10000 (94.37%) lr=0.1
7/20 Test set: Average loss: 9.9793/1.8744, Accuracy: 9446/10000 (94.46%) lr=0.1
8/20 Test set: Average loss: 9.4806/1.9804, Accuracy: 9429/10000 (94.29%) lr=0.1
9/20 Test set: Average loss: 9.1958/1.8623, Accuracy: 9446/10000 (94.46%) lr=0.1
```

```
10/20 Test set: Average loss: 9.0079/1.8324, Accuracy: 9454/10000 (94.54%) lr=0.1
11/20 Test set: Average loss: 8.8913/1.7920, Accuracy: 9483/10000 (94.83%) lr=0.1
12/20 Test set: Average loss: 8.6638/1.7355, Accuracy: 9481/10000 (94.81%) lr=0.1
13/20 Test set: Average loss: 8.3738/1.7513, Accuracy: 9495/10000 (94.95%) lr=0.1
14/20 Test set: Average loss: 8.3480/1.9010, Accuracy: 9459/10000 (94.59%) lr=0.1
15/20 Test set: Average loss: 8.2856/1.7251, Accuracy: 9502/10000 (95.02%) lr=0.1
16/20 Test set: Average loss: 8.0931/1.8259, Accuracy: 9484/10000 (94.84%) lr=0.1
17/20 Test set: Average loss: 8.0200/1.7395, Accuracy: 9514/10000 (95.14%) lr=0.1
18/20 Test set: Average loss: 7.8587/1.7196, Accuracy: 9513/10000 (95.13%) lr=0.1
19/20 Test set: Average loss: 7.8127/1.7602, Accuracy: 9508/10000 (95.08%) lr=0.1
```

In [ ]:

In [ ]: